

Beyond Maximal Random Effects for Logistic Regression: Moving Past Convergence Problems

Amelia E. Kimball^a, Kailen Shantz^a, Christopher Eager^a, Joseph Roy^a

^a *University of Illinois at Urbana-Champaign*

Abstract

Mixed effects models are widespread in language science because they allow researchers to incorporate participant and item effects into their regression. These models can be robust, useful and statistically valid when used appropriately. However, a mixed effects regression is implemented with an algorithm, which may not converge on a solution. When convergence fails, researchers may be forced to abandon a model that matches their theoretical assumptions in favor of a model that converges. We argue that the current state of the art of simplifying models in response to convergence errors is not based in good statistical practice, and show that this may lead to incorrect conclusions. We propose implementing mixed effects models in a Bayesian framework. We give examples of two studies in which the maximal mixed effects models justified by the design do not converge, but fully specified Bayesian models with weakly informative constraints do converge. We conclude that a Bayesian framework offers a practical—and, critically, a statistically valid—solution to the problem of convergence errors.

Keywords: Mixed effects regression, logistic regression, convergence, Bayesian statistical methods, statistics

*Corresponding author: Joseph Roy jroy042@illinois.edu

1. Introduction

The use of mixed effects models has become wide spread across a number of language sciences as a replacement for traditional ANOVA and regression analyses (Baayen et al., 2008; Barr et al., 2013a; Johnson, 2009) . Mixed effects models benefit researchers by allowing participant and item effects to be adequately incorporated into a regression approach through the use of both fixed effects, or traditional regression predictors, and random effects, or the repeated structure of data on participant and item. These are particularly important in language science where variance from participant to participant and item to item is common.

Overall, mixed effects models have proved a useful tool in incorporating subject and item effects. However, these models have a shortcoming: they do not always converge to a solution. At times, lack of convergence is due to the inclusion of a zero or near zero covariance parameter in the mixed effects model, and is an indication that the chosen model is not the most appropriate. At other times, this could reflect an under-powered design. However, convergence errors may also be caused by a number of other factors (see section 1.2), and should not necessarily be taken as a marker that a model is inappropriate. Whatever the cause, convergence errors leave the analyst with a problem: rather than being able to pre-register the model(s) most appropriate for their data and engaging in model comparison, they may need to change their model(s) to address convergence errors. This is particularly problematic because current techniques which address these errors are not well established within the statistical literature on mixed effects models and assume that a lack of convergence is due to mis-specification of the random effects structure (see subsection 1.2). While there is disagreement about how best to determine the random effects structure and we agree with the conclusions of Matuschek et al. (2015), that the “optimal random effects structure” is that which is supported by the data, it remains difficult to assess that optimal structure *a priori*. In addition, as we will discuss, some of the methods that those in the language sciences have proposed for

dealing with convergence issues are problematic from a statistical perspective.

In this paper, we propose a solution for this problem in the form of a fully specified Bayesian mixed effects model, which has been discussed in other venues as a valuable alternative to traditional statistical model in linguistics (e.g. Nicenboim & Vasishth, 2016). This Bayesian implementation of mixed effects places a few more constraints on the original mixed effects model, and in so doing increases the likelihood of convergence.

The goal of this paper is not argue that Bayesian models are the best and only solution for all data sets, nor is it to revisit 50 years of debate between frequentists and Bayesians.¹ Our interest in Bayesian modelling is solely in that such approaches allow researchers to make reasoned constraints on the parameters of interest in order to obtain valid statistical results where the traditional mixed effects model fails to converge.

In this paper, we examine two data sets for which the standard mixed effects models do not converge, whereas the Bayesian implementations converge and thus allow for statistical inference with appropriate random effect structures. We explore convergence issues and argue that a broader Bayesian approach provides one possible solution when an *lme4* mixed effects models approach fails.

1.1. Convergence

Many researchers do not understand why convergence is an issue and what is meant by *convergence*. We present an intuitive discussion below, but refer the interested reader to Monahan (2011) or Lange (2010) for a comprehensive technical overview. In short, in regular ordinary least squares (OLS) regression, finding the estimate of the effect of a predictor involves solving an equation. Specifically, when β is estimated, that estimation process involves maximizing the likelihood for $\hat{\beta}$ and yields $(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$ (in matrix notation). Importantly,

¹ For an outline of the underlying philosophical differences between these two school of statistical thought see Kruschke (2013) or McElreath (2015)

the estimated variance for each predictor is also a closed calculation: $Var(\hat{\beta}) = \hat{\sigma}^2(\mathbf{X}'\mathbf{X})^{-1}$. This closed calculation can be solved by computation, and the only problem that arises is when the data are collinear and thus yield an infinite number of estimates for $(\mathbf{X}'\mathbf{X})^{-1}$.

Through the 1970's a unified approach to non-normal data was developed that resulted in the use of what are termed generalized linear models (GLM) by McCullagh and Nelder (1983). A GLM transforms the non-normal dependent variable via a link function to be a linear combination of the predictors through a singular set of techniques (Hardin & Hilbe, 2007). Prior to the establishment of GLM, there were multiple ways to estimate the regression results for different link functions (see Imrey, Koch & Stokes, 1981 for an overview of pre-GLM techniques for logistic regression).

$$\log\left(\frac{p}{1-p}\right) = \mathbf{X}\beta \quad (1.1)$$

The set of computational techniques used in implementing a GLM model are iterative in order to solve for β and are said to converge when the difference between the current iteration of the algorithm and the previous iteration of the algorithm is less than a pre-defined tolerance. For logistic regression under a GLM, as in equation 1.1, the only time a researcher will run into convergence problems is under what is termed separation. Separation is when there is minimally no variance in the response across a subset of categorical or continuous predictors (e.g. assuming you are modelling accuracy, if all responses for one category are 100% accurate or all observations above a threshold for a covariate are 100% accurate – or 0% in both cases). In this case, it is possible for the regression estimates to separate, or not converge to a solution.

Mixed effects models, on the other hand, require the computation of both the predictor (i.e. fixed effects) estimates and standard errors along with the estimates of Σ , or the covariance matrix for the random effect structure under linear and generalized linear forms. The addition of a random effects structure moves the estimation of the regression estimates (and their standard errors) to

searching for two solutions (one vector of fixed effects and one matrix of random effect variances). Again, the algorithm tests whether or not the difference between the the gradient function of the previous iteration is less than some pre-defined tolerance.

In simple terms, the algorithm iterates between solving for the fixed effects (represented by $\tilde{\beta}$) and Σ , the covariance matrix. The maximum gradient, a function of $\tilde{\beta}$ and Σ from the current and previous iteration, is compared to see if it is less than a set tolerance. If yes, then the model is said to have converged. After this initial test of convergence, the estimates themselves are tested for statistical validity. Even if the model does converge, researchers can still run into the possibility that the estimates are a local maximum, but not a global maximum (i.e. the best estimate for a small area of possibilities, but not the overall best estimate). Demidenko (2013: 96-106) presents the issues surrounding the convergence and statistical validity tests of Σ .

In this paper, the focus is on logistic mixed effects models as they have, in the experience of the authors, more difficulty converging than linear mixed effects models. Intuitively, this could be because while an observation of y for a linear mixed effects model is continuous, for logistic regression each observation represents a binary y . Under logistic regression, each observation is less informative of the relationship between y and the fixed effects especially when controlling for the random effects structure (as each y is only 0 or 1 under logistic mixed effects regression versus a range of continuous possibilities under linear mixed effects regression).

Researchers who are used to the calculations in an traditional regression or who consistently use only one statistical software may not realize that because mixed effect models are implemented with complex iterative algorithms, they differ from software to software. The fact that tolerance settings and computational checks differ from software to software means that a model that achieves convergence in one package may not converge in another package, or even in another version of the same package. Convergence is not a universal truth or a test of the validity of a model, but rather a property of a specific algorithmic

implementation. We do not discuss convergence issues in SPSS or SAS as they are not widely implemented in the language sciences for mixed effects models (see Eddington, 2015 for SPSS implementations of mixed effects models). While the estimation algorithms differ across R, SPSS and SAS, it is difficult to pinpoint precise differences because of the regularity with which these algorithms are tweaked in each platform². It is possible to analyze the same data with these different statistical programming environments and have changes in convergence; it is also possible to have convergence change with different versions of *lme4*. As the *lme4* package represents the standard software implementation of mixed effects models in R in the language sciences, (Baayen et al., 2008; Barr et al., 2013; Gries, 2015) that package will form the basis of this paper.

Bates, Maechler, Bolker & Walker (2015:13-14) lay out some of the computational checks that are used to make sure that the estimates are statistically and computationally valid. As they indicate, these checks are updated and potentially change with different versions of *lme4*. There are currently three convergence checks done in *lme4*³. These checks assess whether the variance estimates are reasonable (i.e. are non-negative and not close to a boundary). There are nine other checks done to ensure that you are following best practice with the data (i.e. scaling the continuous predictors) or not trying to fit more parameters (fixed and random) than you have data for⁴. Any convergence error indicating singularity, a non-positive definite Hessian, or failure for the gradient to converge usually denotes a serious problem with the statistical estimates and requires some sort of manipulation to be applied to the random effects structure to simplify the structure.

²It is even more difficult for SAS and SPSS as they are propriety systems and therefore the implementation of the algorithms are not transparent.

³To see the most up-to-date checks, look in the documentation for `lmerControl()` in *lme4*.

⁴Gelman and Hill (2007:276) do describe scenarios in which it is possible to fit a mixed effects model with random subject intercepts when there is only one observation per subject, but require a fully specified Bayesian model.

1.2. Current approaches to convergence errors

Researchers have debated over what is the optimal default configuration of random effects for analysts, with a maximal random effects structure advocated for by Barr et al. (2013) while more recent work has pushed back against this (Bates, et al., 2015). However, models with any random effects structure, maximal or not, are susceptible to non-convergence. When a model fails to converge, analysts have three options: they may try a different optimizer⁵, increase the number of iterations, or simplify the model. Many language scientists choose to simplify the model. Any number of ad-hoc simplification techniques have been developed to address convergence errors, from a procedural set of reductions (e.g. Jaeger, 2009) to a more formal set of reductions (e.g. Bates et al. 2015). Collectively, the use of these approaches are often reported in results sections as *the maximal random effect structure supported by the data*. We call these methods ad-hoc in the sense that there is a lack of support for these techniques in the statistical literature on mixed effects models. For some approaches, it is unclear if the mathematical properties of the techniques used hold across different data configurations. Other approaches are clearly inappropriate (e.g. (4) and (5) below). Simplification techniques used in the language sciences include the following:

1. (For linear mixed effects models) use a PCA to determine most meaningful slopes. (Bates et al., forthcoming)
2. (For a well designed, well controlled experiment) Reduce item random effect structure then reduce subject random effect structure, following the interaction hierarchy, until convergence (Jaeger, 2009)
3. Use `anova()` to determine the optimal effects structure. (Gries, 2015)

⁵There is an `allFit()` function in the *afex* that attempts to fit all possible optimizers in order to test convergence errors. For the data we present in this paper that process would have taken approximately 60 days with a relatively high speed computer and still not guaranteed that convergence criterion would have been met.

4. Keep removing slopes randomly from item and subject random effect structure until convergence.
5. *Rbrul*, an interactive R program written for sociolinguists, uses *lme4* perform logistic mixed effects models. It began **suppressing all convergence errors** in 2013 and then restored the errors in 2015. (Johnson, 2009)

It is not clear that (1) is applicable to logistic mixed effects models (as presented here) or that what the statistical characteristics of this process yields in terms of optimal random effects structure⁶. The simulation studies presented in Bates, et al (2015) assume that there are three non-zero variances, and show that their method is able to recover them via PCA of the correlation matrix. What is not clear is how such an approach would map to other data configurations or different statistical assumptions made on the correlation matrix (Ω). The approach in (2) could be reasonable if non-convergence could be relied on as indication true zero variance. The use of (3), whether in forward or backward fitting, is addressed several times in the statistical literature on mixed effects (e.g. Demedinko, 2013: 133-137) and, essentially, treating random effects (which are in fact variance components) as if they are fixed effects and calculating model based tests via `anova()` is not a valid model building approach with mixed effects⁷. The problem with such a model building approach is that it treats the random parameters as if they were fixed and penalizes the test statistic for these additional "fixed" parameters. Even evaluating the statistical significance of the fixed effects parameters in a mixed effects model is not a straightforward task (see Luke, 2016 for discussion). The last two approaches, (4) and (5), are understandable when researchers who have no training in statistics confront convergence errors and do not comprehend why they invalidate the results and

⁶There are very real differences between a linear and generalized mixed effects model that even statisticians who are not familiar with the mathematical structure of these models misinterpret (see Molenberghs & Verbeke, 2005:297-301).

⁷If there is a known fixed effects structure, it is less problematic to reduce the random effects in this way, but there is not a published study we are aware of in which the researcher already knows what the fixed effects, their magnitude and variance are.

how to address them. It should be fairly obvious that ignoring convergence errors or removing slopes at random is not sound statistical practice.

In this paper, we present a Bayesian approach to dealing with convergence issues with mixed effects models using the maximal random effects structure. In order to understand why the use of Bayesian modelling in order to preserve the maximal random effects structure is useful, though, it is necessary to first understand why having a maximal random effects structure is desirable. Here, it is important to recognize that, like ANOVAs, mixed effects regression models assume independence of observations. For ease of exposition, we will discuss what this means in the context of an experimental study in which a researcher directly manipulates an independent variable to observe the effects of different experimental conditions on a dependent variable of interest. Note, however, that the following discussion applies equally to observational studies in which an independent variable is not directly manipulated.

For between-subject designs, the assumption of independence of observations across conditions holds so long as a subject only encounters one experimental condition. For research using within-subject designs, however, this assumption is not met. This is because the observations for a particular subject in one condition are not independent of the observations for that same subject in a different experimental condition. In such a case, it is necessary to include a by-subject random slope for condition in order to satisfy the assumption of independence of observations. Similarly, if the same item occurs in different experimental conditions, a by-item random slope for condition must be included to satisfy this assumption. Without properly satisfying this assumption, a researcher will ultimately arrive at an invalid model (for more on this issue, see Barr et al.s, 2013 discussion of conditional independence). Empirically, it has also been demonstrated that failure to include random slopes for predictors of interest results in greater Type I error rates (Barr et al., 2013b; Schielzeth & Forstmeier, 2009; see also Matuschek et al., 2015). Barr (2013) has further shown that Type I error rates are inflated when higher order interactions including a predictor of interest are not included as random slopes. Thus, there are both theoretical and

empirical reasons to include appropriate random slopes for all main effects and interactions involving predictors whose effects we are interested in estimating; when a model does not converge and a researcher chooses to simplify the random effects structure, they consequently run the risk of inflating Type I error due to a violation of conditional independence, in addition to which they risk inflating Type I and Type II errors due to increased researcher degrees of freedom in deciding how to simplify the model. Thus, there is good reason for a researcher to include a maximal random effects structure and to avoid simplifying that structure if at all possible.

Finally, we acknowledge that power is important and that *enough* data need to be gathered for the random effects structure in line with Matuschek et al. (2015). In practice, however, this advice becomes problematic if there is a failure for the researcher’s maximal random effects structure to converge: then, and only then, do researchers consider the possibility of the design and sample being an *under-powered* study. First, lack of convergence does not necessarily mean that a researcher’s design and sample are under-powered. More importantly, how does a researcher move forward from lack of convergence if there are theoretically compelling reasons to include the maximal or near-maximal random effects structure in the model? Adding more subjects may not solve this problem and in fact violates best practices with respect to stopping rules. Moreover, *more data* might not lead to convergence if, in fact, it is something other than small slope/intercept variances or model misspecification causing non-convergence⁸.

⁸As an example, the third author has a phonetic data set that includes 90 subjects and more than 65 thousand observations: the maximal random effects structure does not converge for this data set, but there is strong theoretical motivation to believe that there is non-zero variance in each of the random slopes which is backed up by the descriptive statistics. The last two authors are currently working on a project that demonstrates that this common assumption does not hold under certain simulated conditions

1.3. Mathematical Specification

Many researchers use mixed effects models without attention paid to the formal mathematical model, but in order to better understand what a fully specified Bayesian model is and how it differs from the traditional mixed effects model, a discussion of the formal mathematical model is necessary. The traditional linear mixed effects model is formally defined below:

$$\mathbf{y} = \mathbf{X}\beta + \mathbf{Z}\mathbf{u} + \epsilon \quad (1.2)$$

The observed vector, \mathbf{y} , is the response variable of interest and the \mathbf{X} is the matrix of coded fixed effects. β represents the fixed effects. \mathbf{Z} is the design matrix of random effects and we have $\mathbf{u} \sim N(0, \Sigma)$ where Σ represents the covariance matrix for the random effects⁹. What is important to retain from this definition is that no constraints are placed on β and no constraints are placed on Σ .

The lack of constraints carries over into a logistic mixed effects regression that is simply a linear mixed effects model which has been transformed via the logistic link function into:

$$\log\left(\frac{p}{1-p}\right) = \mathbf{X}\beta + \mathbf{Z}\mathbf{u} + \epsilon \quad (1.3)$$

The parameters estimated via *lme4* are β and Σ . The random effects structure, \mathbf{u} , is considered an ancillary parameter in traditional formulation of this model (Molenberghs and Verbeke, 2005) in that the parameter of interest is β . Again, implicit in this definition is the lack of distributional specification of β and Σ . More specifically, for example, a β of -13 would be equally as likely as a β -3, regardless of the experimental design. For a categorical predictor to have a true log-odds of -13, however, would be quite remarkable and would mean

⁹There is another component that is often left undiscussed in the linguistics literature on mixed effects models, namely the possible covariance structures for ϵ which are termed R-side effects in the statistical literature, and can capture variance structures that are more time-series based—e.g. moving averages

the probability of that event being .0000022 (c.f. a true log-odds of -3, which would be .05). In effect, an unconstrained estimate allows for effect sizes to be overestimated.

1.4. A Bayesian Approach

In contrast to standard mixed effects models with *lme4*, a Bayesian approach requires the selection of constraining distributions on the fixed effects (and random effects) to regularize the estimates to a more reasonable set of possibilities. In Figure 1, one possible constraint distribution for the fixed effects is displayed. Traditional regression would presuppose that any estimate in $(-\infty, \infty)$ is reasonable and equally likely. In logistic regression, the estimates represent the change in log-odds for a predictor: a change from -4 to 4 represents a change from 2% probability to 98% probability. The mathematical effect of choosing this constraint is to regularize the regression estimates. Rather than assuming that an extremely large log-odds (e.g. -14 or 14) is likely, this constraint weighs estimates near zero as more likely (see Gelman et al. 2014a). The assumption against large effects helps prevent errors in overestimating the magnitude of the effect.

Mixed effects models can be thought of as hybrid Bayesian-Frequentist models (Demidenko, 2013) in that they add more variance components with separate distributional assumptions to the standard error present in ordinary least squares regression (e.g. a variance component for participants, items, schools or any other unit of repeated measurement). Bayesian approaches extend the mixed effects model implemented in *lme4* with constraints on the fixed and random effects parameters (that is, the β estimates and the estimates of Σ and Ω). Within Bayesian terminology, these would be *priors*, but we avoid that term here because with mixed effects models these distributions simply represent a more strict set of assumptions. A linguistic introduction to Bayesian

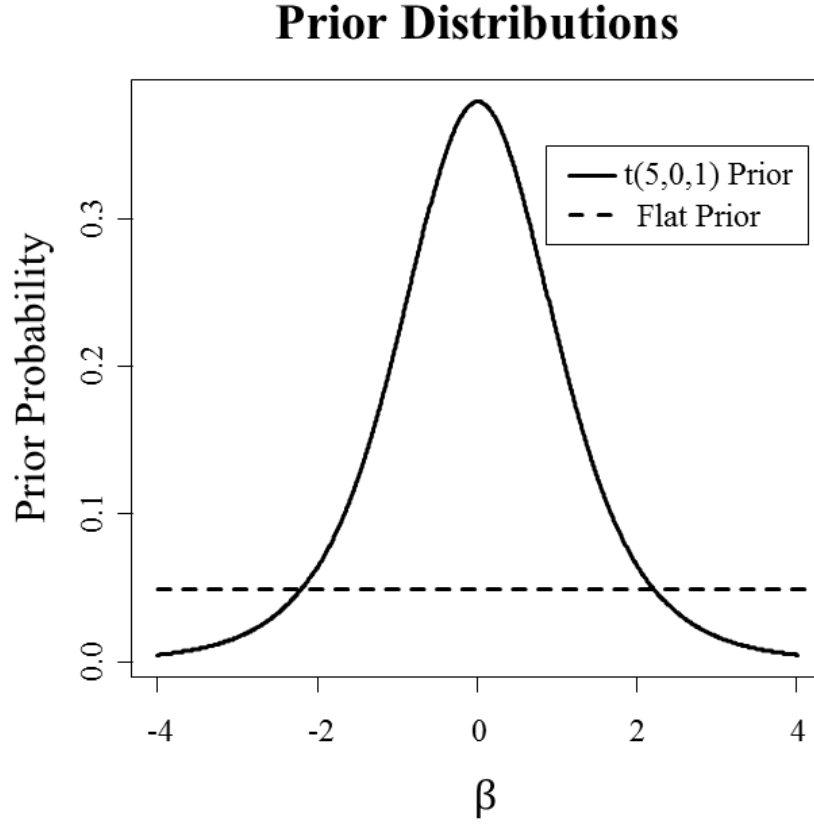


Figure 1: Density plot of a t-distribution with 5 df

approaches statistics is in Nicenboim and Vasishth (2016).

$$\text{Posterior} \propto \text{Prior} \times \text{Data} \tag{1.4}$$

There is an informal representation of the Bayesian approach to statistics in Equation 1.4: The outcome (posterior) is a function of the model constraints (the prior) and the observed data. The less data or the more noisy the data, the more the prior constraints on the model matter.

The most controversial aspect of a Bayesian approach is that it requires constraints¹⁰ be placed on the regression estimates (fixed and random) that are not used in a more traditional approach to mixed effects. For example, in the present paper we assume that the fixed effects for both data sets we examine below are constrained by a t-distribution rather than a uniform distribution as assumed by mixed effects models. The complicated models set up by a fully specified Bayesian model cannot be computed directly and require computationally complex methods to estimate the results (see Gelman et al., 2014a, or Kruschke, 2013 for an introduction to these methods). The data can also overwhelm the prior – if there is evidence that the prior distribution is incorrect, this will be illustrated in the posterior distribution of your estimates. The weakly informative priors selected in this study serve to guide the estimation process in a reasonable direction while not predetermining the results of the model. A somewhat imperfect analogy is that if you lose your keys in Chicago, it is not helpful to start looking for them in Montreal. Similarly, we start by assuming most experimental effects are small and within a moderate window around 0 while also allowing for large effects to manifest, if there is sufficient evidence and coverage in the data.

There is still the possibility for non-convergence in a fully specified Bayesian model, especially if the priors are severely misspecified. Gelman, et al. (2006: 281-286) discuss assessing convergence for these models. Convergence can be assessed through three main measures: R-hat, effective sample size, and the number of divergent transitions that occur post-warmup. The R-hat (ibid: 285) metric is a ratio of the variance of our model to the within sequence variance. Convergence is indicated when R-hat values are less than 1.1. The effective sample size is a measure of the precision of our parameter (fixed and random effect)

¹⁰These typically are referred to as prior distributions, denoting some *prior* belief about the distribution of the data. We avoid that terminology as there are a number of motivations for these constraints – some based on prior knowledge and some, discussed in this paper, based on computational considerations.

estimates (ibid: 287) and ideally should be at least 100. Divergent transitions occur when the algorithm’s stepsize is too large and a computation which is finite theoretically diverges to infinity, and so post-warmup (when the stepsize becomes fixed for sampling), there should ideally be no divergent transitions. This issue of divergent transitions for these models can often be remedied by increasing the delta parameter and re-running the Bayesian model in Stan¹¹.

2. Method and Data

In this paper we give two examples of mixed effects models with binary dependent variables. We believe that these are representative examples of typical language experiments common in psychology and linguistics that are best modeled with logistic regression. In both cases, previous research provides strong justification for a limited set of predictors and interactions, but convergence errors create problems for the analyst.

2.1. Behavioral data from a psycholinguistic study

Dataset one comes from the accuracy data reported in Shantz & Tanner (2016). Their study investigated the relative timing with which grammatical gender and phonological information are retrieved during lexical access. Early work using event-related potentials (ERPs) to investigate lexical access had found that grammatical gender is retrieved prior to phonological information (Van Turennout, Hagoort & Brown, 1998). Subsequent research, however, has shown task effects on the relative time course of lexical access (Abdel Rahman, Van Turennout & Levelt, 2003; Abdel Rahman & Sommer, 2003) as well as evidence for specific effects of task order (compare Van Turennout, Hagoort & Brown, 1997 to Schmitt, Münte & Kutas, 2000; see also Gomez, Ratcliff & Perea, 2007) on lexical access. In light of these more recent findings, Shantz & Tanner (2016) combined the use of ERPs with the dual-choice go/no-go paradigm in

¹¹This process usually increases the amount of the model takes, but no model for this paper ran more than 10 hours

order to examine how task order impacts the relative timing with which grammatical gender and phonological information are retrieved during lexical access.

In their experiment, twenty native speakers of German were presented with 24 black-and-white images depicting high frequency, concrete German nouns with high naming agreement. Nouns differed orthogonally in their grammatical gender and word-initial phone. On individual trials, participants were presented with an image and asked to make a set of decisions based on the grammatical gender and the phonology of the depicted noun. One source of information (e.g. gender) was used to decide whether or not to respond (i.e. the go/no-go decision), and the other source of information (e.g. phonology) determined whether to respond with the left or right hand (i.e. the dual-choice decision). The mapping of information (i.e. gender or phonology) to decision (go/no-go or dual-choice) and response hand were fully counterbalanced within subjects; each possible configuration occurred in a separate experimental block. Within a block, items were presented four times each, yielding a total of 768 trials per participant. To manipulate task order, the order of blocks was counterbalanced across four lists: two in which go/no-go decisions were based on phonology for the first half of the experiment and gender for the second half, and two in which go/no-go decisions were determined by gender for the first half of the experiment and phonology for the second half. Ten participants were assigned the gender = go/no-go first task order, and the other ten had the phonology = go/no-go first task order.

In addition to task order, condition (i.e. hand = phonology or hand = gender), trial type (i.e. go or no-go) and the interactions between these variables were the predictors of primary interest. Trial type and condition were included in the accuracy model as these were the two variables on the basis of which relative time course information was determined in the electrophysiological data. It was thus important to determine whether task order would impact accuracy as a function of these predictors. The logistic mixed effects regression model fit to this data also included a number of control predictors. Trial was included in the model given that performance in psycholinguistic experiments can change over

the course of a session due to learning (e.g. Fine, Jaeger, Farmer & Qian, 2013) or fatigue. Thus, including this predictor controlled for any systematic change in accuracy over the course of the experiment. Grammatical gender was included as a control predictor, as prior research has shown effects of grammatical gender on lexical access (Akhutina, Kurgansky, Polinsky & Bates, 1999; Bates, Devescovi, Hernandez & Pizzamiglio, 1996; Opitz & Pechmann, 2016). Because phone frequency has been found to influence errors in speech production (Levitt & Healy, 1985), and because participants were required to retrieve information about each item’s initial phone, the initial sound was further included as a control. Word frequency, moreover, has known effects in lexical access (Jescheniak & Levelt, 1994; Strijkers, Costa & Thierry, 2009; Strijkers, Holcomb & Costa, 2011), and was therefore included in the model. Finally, the number of syllables in a word has also been shown to influence production in picture naming paradigms (Alario et al., 2004); syllable count was thus included as a control predictor. Because the task order manipulation split conditions across different halves the the experiment for each group, any interaction between condition and task order was potentially confounded by trial effects. The authors thus also included the three-way interaction of task order x trial x condition and its subordinate interactions to statistically control for this potential confound. Table 1 summarizes the predictors included in the full model for Shantz & Tanner (2016) prior to model simplification.

2.2. *Perception Study*

The second study we examine is data from in Kimball & Cole (unpublished results) . This study investigates the effects of phonological features and phonetic detail on the reported perception of stressed syllables in sentences of English. Recent experiments on prosody perception show that listeners use both signal-based cues (e.g. duration) and top-down cues (e.g. information structure) when reporting perceived prominence of words in English (Cole et al., 2010b; Bishop, 2012). However, it is not known whether a similar array of factors influence perception of syllable stress. Perception of syllable stress has

Table 1: Summary of Predictors for Dataset 1

Categorical Predictors			
Predictor	Levels	Variable Type	
Task Order	1.Phonology = Go/No-Go First 2.Gender = Go/No-Go First	Dichotomous	
Trial Type	1.Go 2. No-Go	Dichotomous	
Condition	1. Hand= Phonology 2. Hand =Gender	Dichotomous	
Grammatical Gender	1. Masculine 2. Neuter	Dichotomous	
Initial Sound	1. /k/ 2./b/	Dichotomous	
Syllable Count	1.One 2. Two 3. Three	Ordinal	
Continuous Predictors			
	Range	Mean	s.d.
Trial	1-768	384.500	221.847
Log Frequency	1-3.916	1.458	0.644
Interactions			
Task Order x Condition			
Trail Type x Condition			
Task Order x Trial Type x Condition			
Task Order x Trial x Condition			
Random Effects Structure			
Intercept Term	Slope Term		
Participant	Condition		
	Trial Type		
	Condition x Trial Type		
Item	Condition		
	Task Order		
	Trial Type		
	Condition x Task Order		
	Condition x Trial Type		
	Task Order x Trial Type		
	Task Order x Trial Type x Condition		

important consequences for comprehension, because prior work shows that metrically regular patterns affect linguistic processing in detection tasks (Quené & Port, 2005; Zheng & Pierrehumbert, 2010), as well as in production tasks (Tilsen, 2011), in memory for heard speech (Kimball, Yiu, & Watson, unpublished results), and even in silent reading (Breen & Clifton Jr., 2011). Kimball & Cole report on a stress perception study that tests the interaction of acoustic cues with top-down cues related to metrical context and lexical stress location.

For the experiment, self-reported native speakers of American English were recruited online using Amazon Mechanical Turk. Kimball and Cole use a metalinguistic stress reporting task in which participants are presented with an audio file along with a syllable-by-syllable transcription of that file, displayed in a browser window using Qualtrics survey software. Participants listened to recordings of sentences and marked the stressed syllables in a transcript of the sentence provided for them. The listeners annotated re-recorded sentences from the Buckeye corpus (Pitt et al., 2007) that were produced by a trained linguist (not a member of the research team), as well as individual words and a poem with a distinctive rhythmic pattern produced by the same speaker. In total, 94 subjects marked 403 syllables each for a total of 37,882 data points.

Kimball and Cole analyzed their data with a mixed effect logistic regression, with a dependent variable of whether the syllable was marked as stressed or not. Syllable, mean F0, intensity, and duration were chosen as signal-based factors based on work that suggests these are acoustic markers of prosodic prominence (Cole et al., 2010b; Breen et al., 2010; Bishop, 2012). For non-signal-based features, stress location in citation form (i.e. primary stress as it would be marked in the dictionary) was chosen based on previous work by the authors which suggested this was a reliable predictor of reported syllable stress (Kimball & Cole, 2014). Additionally, function/content word was included as a predictor because function words are known to be less likely to attract stress (Selkirk, 1995).

Lastly, metrical context was included because prior work on stress shift (Vogel et al., 1995; Grabe et al., 1995) indicates that the metrical structure of

Table 2: Summary of Predictors for Dataset 2

Categorical Predictors			
Predictor	Levels	Variable Type	
Stress	1.primary stress	Dichotomous	
	2.no primary stress		
Function word	1.Yes	Dichotomous	
	2. No		
Word to the left marked	1. Yes	Dichotomous	
	2. No		
Word to the right marked	1. Yes	Dichotomous	
	2. No		
Experiment instructions	1. “mark the beat”	Dichotomous	
	2.“mark the syllable”		
Continuous Predictors			
	Range	Mean	s.d.
F0(Hz)	77.41-372.47	177.14	37.840
Intensity (dB)	47.44-76.20	64.22	4.685
Duration (ms)	40.8-848.0	277.4	113.994
Interactions			
F0 x Intensity			
F0 x Duration			
Duration x Intensity			
F0 x Duration x Intensity			
Random Effects Structure			
Intercept Term	Slope Term		
Subject	Stress		
	Function word		
	F0		
	Intensity ₂₀		
	F0xIntensity		
Item	Experiment		
	Previous word marked		

English is structured such that two stresses next to each other (a “clash”) is dispreferred. This was incorporated in the model by including a predictor indicating whether the previous syllable was marked. The work presented here was originally run as two separate experiments with slightly different instructions but identical procedures; results of the two experiments are pooled in this model, and so experiment is added as a predictor. Based on previous research, variability between subjects and items is expected on all of the predictors except experimental instruction (Kimball & Cole, 2014; Cole et al., 2010a), and so there is strong motivation to create the maximal model including both random slopes and intercepts. Table 2 summarizes the predictors included in the model reported by Kimball & Cole.

2.3. Model Fitting Procedure

For the data from the two experiments, we extend the weakly informative constraints of Gelman et al. (2013: 412-420) for logistic regression to a mixed effects model. For the regression results presented in this paper, the weakly informative priors (1)-(6) are listed below.¹²

1. $\beta \sim t(5, 0, scale)$
2. $scale \sim \text{Half-Normal}(0, 1)$
3. $\sigma_{RE} \sim \text{Half-Normal}(0, 1)$
4. $\gamma_{RE} \sim N(0, \Sigma)$
5. $\Sigma = \text{diag}(\sigma_{RE}) \Omega [\text{diag}(\sigma_{RE})]^T$
6. $\Omega \sim \text{lkj}(\eta = 2)$

For logistic regression, (1) and (2) are important constraints that encode several reasonable assumptions about the regression results (see Gelman et al., 2008 for a version of this constraint and Gelman et al., 2014b: 412-417). Under a standard logistic regression (mixed or not), the *a priori* assumption is that

¹²R Stan was used to generate the statistical results presented in this paper (Carpenter et al., 2016). The Stan code and model code is provided in Appendix A.

the fixed effects can vary uniformly from $(-\infty, \infty)$. The assumption in (1), intuitively, states that if a researcher were to repeat the experiment a number of times, the expected fixed effects estimate distribution would be a bell-shaped curve centered at 0, with more probability in the tails than a normal distribution of the same scale (in other words, we are allowing for a higher probability of large effects than a normal distribution, but we still expect small or null effects to be more likely). Rather than determining the scale of this distribution *a priori*, we also make the scale a parameter to be estimated. The half-normal prior in (2) has a mean of approximately 0.8, considerable probability density between values of 1.0 and 2.0 (probability approximately 0.27), and less for values greater than 2.0 (probability about .05). In any given logistic regression of any given research design, an effect estimate of 100 is not equally likely as 1. In a logistic mixed effects model, there is only the assumption of the random effects, (4), being multivariate normally distributed. The additional constraint (3) acts on the standard deviations in the random effects in the same way the half-normal prior acts on the scale of the fixed effects. The additional constraint (6) on the correlation matrix of the random effects, Ω , is discussed in detail in Flaxman et al. (2015). The implementation of these constraints is reparameterized for computational efficiency as discussed in Carpenter et al. (2016).¹³

3. Results

3.1. Behavioral task

Results from Shantz & Tanner (2016) found a robust effect of task order (i.e. whether the go/no-go decision was determined by gender or by phonology first); the group of participants who used phonology to make the go/no-go decision in the first half of the experiment were significantly faster at responding, and showed electrophysiological evidence for earlier retrieval of grammatical gender

¹³All continuous predictors were scaled prior to analysis (i.e. centered and divided by their standard deviation), sum contrasts were used for unordered categorical predictors, and orthogonal polynomial contrasts were used for ordered categorical predictors.

over phonology, consistent with prior research (Van Turennout et al., 1998). In contrast, the gender = go/no-go first group showed no evidence for earlier retrieval of gender. While the accuracy data are numerically consistent with this trend, showing higher accuracy for the phonology = go/no-go first group, the mixed effects model fit to this data found no reliable main effect of task order. Importantly, however, this model did not converge with the maximal random effects structure due to a warning about the Hessian¹⁴. In fact, the model reported in Shantz & Tanner (2016) was simplified down to only a random intercept by subject, and even then did not converge.

Given that participants were overall highly accurate in the experiment, whether or not task order affects accuracy would ultimately not change the results of the study or the interpretation put forward by the authors. The conclusion that there is no main effect of task order on accuracy is, however, in light of the non-convergence, based on unreliable model estimates and an assumption that there is no random variance by items nor by subjects. A Bayesian analysis can thus help to obtain reliable model estimates with an optimal random effects structure in order to confirm or disconfirm whether task order influences participants' accuracy.

The Bayesian model fit to this data included all of the same fixed effects as in the original model, as well as the maximal random effects structure, which included appropriate slopes for all fixed effects parameters that varied within-subjects or within-items. The reported model was fit using weakly informative priors. Convergence was assessed by visual inspection of the traceplots and by examining the \hat{r} values.

¹⁴Using different optimizer algorithms (see appendix C) also failed to produce a model that converged on the maximal random effects structure.

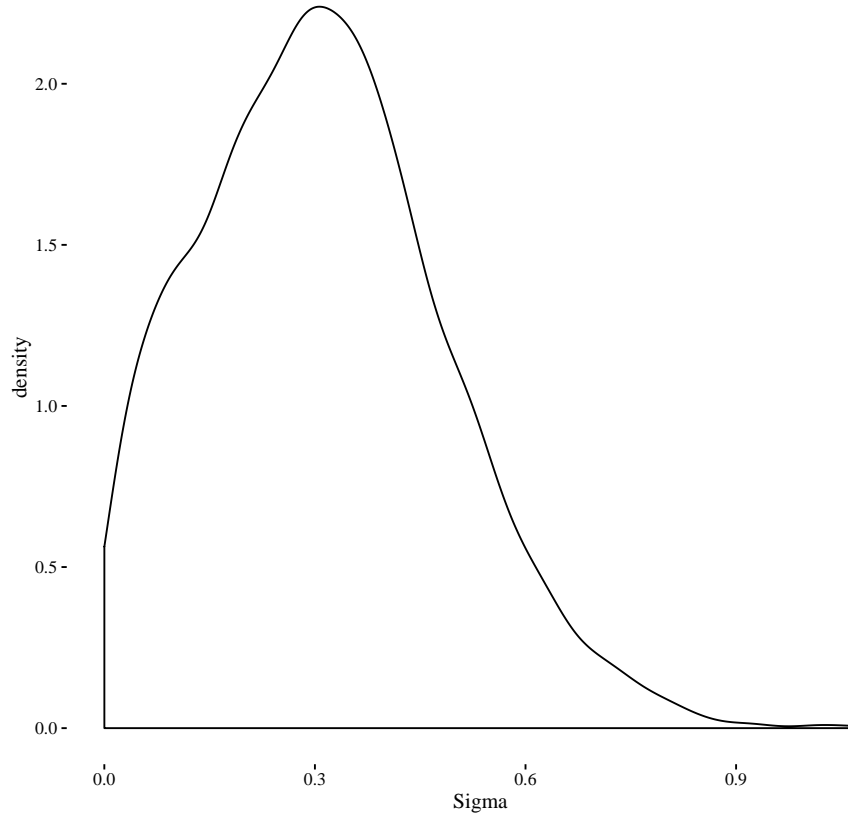


Figure 2: Density plot of the variance estimates for the random slope of trial type by subjects produced by the Bayesian model

The variance components (i.e. the random effects) were examined by plotting density functions of the variance estimates from the Bayesian model. If a random effect truly has no variance, we would expect the variance estimates in its density plot to be tightly clustered near zero, and for the peak of the density function to be at zero. Figure 2 shows the variance estimates of the random slope for trial type by subject. Plots of the variance estimates for all other random effects can be found in Appendix B.

In all cases, there is good evidence for non-zero variance in the random

effects. Thus, the failure of the standard model fit with *lme4* to converge is not likely attributable to a lack of variance by items or by subjects.

The results for the standard mixed effects model fit with *lme4* and for the Bayesian model fit with *rstan* are summarized in Table 3.¹⁵ The directions of any effects are not discussed, as these are incidental to the goal of this paper. Rather, discussion centers on critical similarities and differences across models in effect sizes and measures of significance (or in Bayesian terms, the strength of the evidence for an effect) for the predictors of interest.

¹⁵Note that the exact estimates provided in this table differ somewhat from those reported in Shantz & Tanner (2016). This is because we refit the model reported by Shantz & Tanner using sum coding for categorical variables and scaled continuous variables in order to make the results directly comparable to the output of the Bayesian model. Importantly, the general pattern of results remain the same. Moreover, while the model we fit does actually converge for subject intercepts only, it still fails to converge for any more complex random effects structures.

Table 3: Summary of model results for the behavioral study using a standard glmer and a Bayesian model

Predictor	Standard glmer				Bayesian Model				
	Parameter estimates		Wald's test		Parameter estimates		credible interval		$p(\hat{B} > 0)$
	B	S.E.	z	pz	B	S.E.	2.5%	97.5%	
Intercept	4.86	0.25	19.67	<0.001	5.14	0.01	4.61	5.72	1.00
Trial	0.24	0.14	1.69	0.092	0.23	0.00	-0.05	0.52	0.94
Trial Type	-0.65	0.09	-7.26	<0.001	-0.67	0.00	-0.99	-0.36	0.00
Task Order	0.25	0.22	1.16	0.247	0.14	0.00	-0.21	0.53	0.77
Condition	0.38	0.15	2.49	0.013	0.20	0.00	-0.12	0.57	0.88
Gender	-0.01	0.07	-0.16	0.875	-0.03	0.00	-0.27	0.20	0.41
Initial Sound	-0.00	0.08	0.00	1.00	0.06	0.00	-0.18	0.30	0.70
Frequency	0.14	0.11	1.32	0.186	0.03	0.00	-0.23	0.30	0.60
Syllable Count (Linear)	0.32	0.29	1.10	0.270	0.11	0.01	-0.36	0.66	0.67
Syllable Count (Quadratic)	0.45	0.17	2.61	0.009	0.28	0.00	-0.10	0.71	0.92
Task Order x Trial	-0.32	0.14	-2.23	0.026	-0.21	0.00	-0.53	0.06	0.07
Task Order x Trial Type	0.04	0.09	0.49	0.627	0.05	0.00	-0.18	0.29	0.67
Trial x Condition	-0.05	0.14	-0.37	0.711	0.03	0.00	-0.24	0.32	0.58
Trial Type x Condition	-0.17	0.09	-1.89	0.060	-0.08	0.00	-0.35	0.20	0.28
Task Order x Condition	0.03	0.15	0.21	0.832	-0.03	0.00	-0.34	0.27	0.42
Task Order x Condition x Trial	-0.02	0.14	-0.17	0.867	0.06	0.00	-0.23	0.37	0.66
Task Order x Condition x Trial Type	-0.19	0.09	-2.11	0.035	-0.14	0.00	-0.42	0.12	0.14

Note: Bolding denotes predictors where there was a change in the interpretation of whether an effect was reliable or not

Results for the glmer show main effects of trial type and condition but no main effect of task order, nor any interactions between task order and condition or task order and trial type. The model does, however, find a significant three-way interaction between task order, trial type and condition, as well as a marginal interaction between trial type and condition that was fully significant in the model reported by Shantz & Tanner (2016). Because accuracy was not of primary interest to Shantz & Tanner (2016), and because the authors knew any conclusions would ultimately be based on unreliable estimates from a sub-optimal model, they refrained from interpreting these results other than to suggest on the basis of prior literature that the apparent effect of trial type may reflect the fact that no-go trials only required deciding whether or not to press a button, whereas go trials required also deciding what button to press.

Comparing the results of the glmer to those of the Bayesian model it is immediately evident that Shantz & Tanner’s caution in interpreting their accuracy results was well justified. In contrast to the glmer, the Bayesian model finds no evidence for an interaction between condition and trial type. The credible interval for this parameter contains zero, and the posterior probability that the parameter is different from zero in the direction of the beta’s sign is close to chance. The Bayesian model also finds very limited evidence for the three-way interaction between condition, trial type and task order. The posterior probability that this parameter is less than zero is 86%, however the credible interval for this parameter nonetheless contains zero, for which reason we cannot be confident that there is a real effect. Similarly, the Bayesian model finds only weak evidence for an effect of condition, with an 88% posterior probability that the effect of this parameter is greater than zero, but with zero contained in the credible interval.

Though the Bayesian model arrives at different results than the standard glmer for some parameters, it does reveal similar results for others. Consistent with the glmer, the Bayesian model finds strong evidence for an effect of trial type, as indicated by the fact that the credible interval does not contain zero. There is, accordingly, a 100% posterior probability, given the data, that there

is an effect of trial type. The Bayesian model further finds no clear evidence for an effect of task order: the credible intervals for the main effects and the task order x condition and task order x trial type interactions all contain zero and the posterior probabilities that the beta estimates for these parameters are different than zero are close to chance (i.e. 50%). The standard model and the Bayesian model thus yield similar results for a number of the predictors of interest. Nonetheless, it is important to remember that any conclusions drawn from the glmer are based on unreliable estimates due to its failure to converge with a more optimal random effects structure; any conclusions must therefore be made with caution.

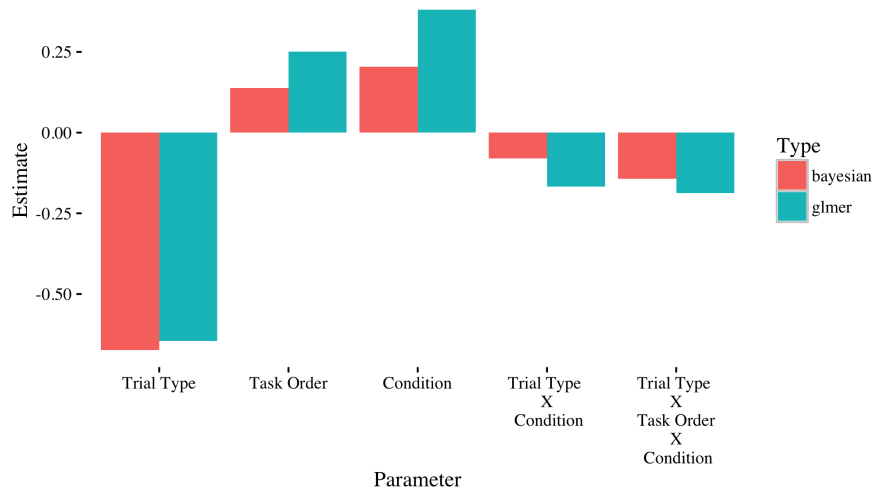


Figure 3: Plot of the beta estimates from the glmer and the Bayesian models for Task Order, Trial Type, and the interactions between Condition and Trial Type, and between Condition, Task Order and Trial Type

In addition to providing different results about which parameters have reliable effects on accuracy, the model comparison further reveals that the glmer tended to overestimate effect sizes. This is illustrated in Figure 3 for the three predictors of interest that reached significance in the glmer, the main effect of

task order, and the interaction between trial type and condition. Thus, the model comparison has shown that the failure of the standard glmer to converge with an optimal random effects structure invites two spurious conclusions: first, that task order does indeed have an effect on accuracy in a three-way interaction with condition and trial type; and second, that many significant effects in the model are larger than the more optimal model indicates. Because, however, the Bayesian model was able to converge on a random effects structure that takes into account by item and by subject variance for the predictors of interest, it is able to minimize Type I errors and provide more reasonable effect size estimates. As one final point of consideration about what these modeling results have revealed, note that the lack of any evidence for an effect of task order in the Bayesian model means that a Bayesian approach would have permitted Shantz & Tanner (2016) to reasonably conclude that task order had no effect on accuracy in their data. As previously noted, this would not have changed the main conclusions reached in their experiment. It does, however, indicate that the effect of task order on lexical access may be restricted to the time course of information retrieval without increasing the likelihood of retrieval errors. Thus, this analysis indicates that a Bayesian approach to overcoming convergence issues can help both to prevent researchers from drawing spurious conclusions and to permit researchers to draw conclusions that are actually justified by the data.

3.2. Perception study

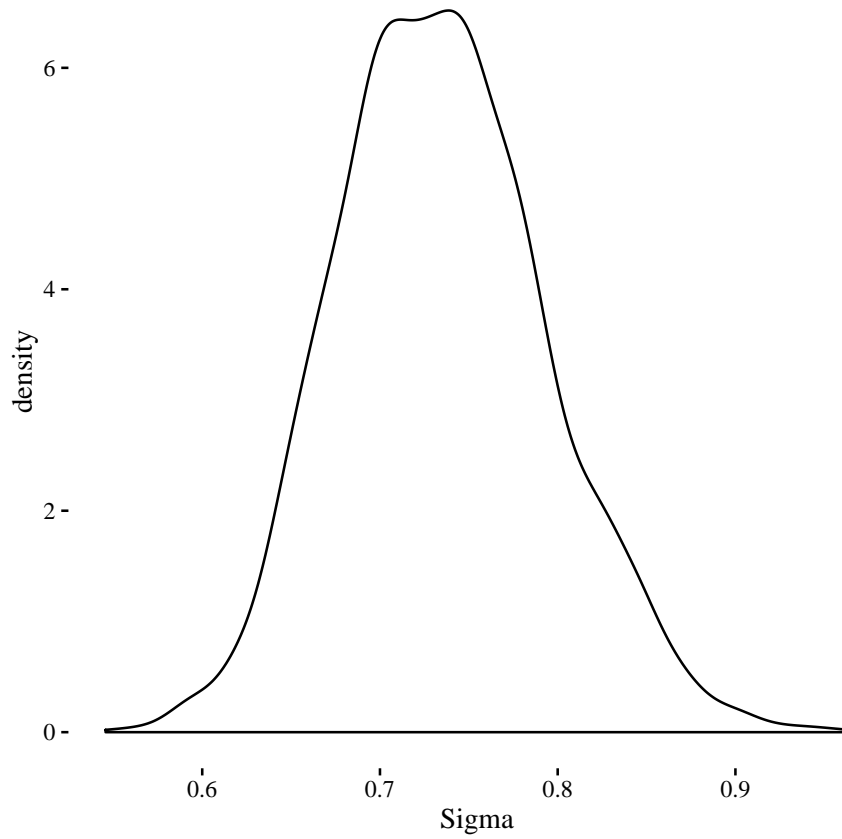


Figure 4: Density plot of the variance estimates for the random slope of primary stress by subjects produced by the Bayesian model

Results of the glmer indicate that all predictors are significant, with the exception of the interaction of F0 and duration and the three-way interaction of F0, intensity, and duration. The results were consistent with Kimball and Cole's theoretical predictions. However, as was the case for the behavioral task reported above, the model did not converge with the maximal random effects

structure¹⁶. Indeed, convergence was not achieved until the model was simplified to include only a random intercept by subject and no other random slopes or intercepts. In other words, the glmer model did not account for variability in baseline effects by item or for variability within subjects and items. Variability by items as well as variability in effects within subjects is expected based on previous research (e.g. Kimball & Cole, 2014; Cole et al., 2010a), and so leaving out these intercepts and slopes is particularly problematic.

As in the behavioral study, the variance components (i.e. the random effects) were examined by plotting density functions of the variance estimates from the Bayesian model. For example, Figure 4 shows the variance estimates of the random slope for primary stress by subject. Plots of the variance estimates for all other random effects can be found in Appendix B. As in the behavioral study, for all predictors the variance plots show evidence for non-zero variance. Thus, the lack of convergence with *lme4* is most likely not due to low or zero variance by items or by subjects.

In contrast, the Bayesian model fit to this data included the maximal random effects model justified by the design, including random slopes for subjects and items. The reported model was fit using weakly informative priors (Gelman & Hill, 2007). Convergence was assessed by visual inspection of the traceplots and by examining the \hat{r} values.

¹⁶The model failed to converge with a max gradient of .006. This is close enough to a .002 threshold that allowing the algorithm to estimate for, say, ten times longer might have led to convergence. However, the model specified here took 1 day to run. From a practical standpoint, we believe that advocating for models which would take ten days to run would be unreasonable when there is a solution in the form of Bayesian approaches. We also tried many different optimizers, as outlined in Appendix C, none of which achieved convergence.

Table 4: Summary of model results for the perception study using a standard glmer and a Bayesian model

Predictor	Standard glmer				Bayesian Model				
	Parameter estimates		Wald's test		Parameter estimates		credible interval		$p(\hat{B} > 0)$
	B	S.E.	z	pz	B	S.E.	2.5%	97.5%	
Intercept	-1.90	0.15	-12.85	<0.001	-1.82	0.00	-2.11	-1.52	0.00
Primary Stress	-0.17	0.11	-1.52	0.129	-0.13	0.00	-0.34	0.08	0.11
Function Word	0.90	0.09	10.01	<0.001	0.86	0.00	0.69	1.04	1.00
Experimental Instructions	0.16	0.08	1.86	0.062	0.16	0.00	-0.02	0.33	0.96
Previous Word Marked	0.64	0.11	6.04	<0.001	0.60	0.00	0.40	0.82	1.00
F0	0.19	0.06	2.96	<0.001	0.17	0.00	0.06	0.30	1.00
Intensity	0.38	0.07	5.49	<0.001	0.38	0.00	0.24	0.51	1.00
Duration	0.73	0.08	9.59	<0.001	0.72	0.00	0.561	0.87	1.00
F0 x Intensity	0.17	0.05	3.19	<0.001	0.16	0.00	0.05	0.27	1.00
F0 x Duration	-0.01	0.08	-0.08	0.937	-0.01	0.00	-0.16	0.14	0.43
intensity x Duration	0.33	0.06	5.41	<0.001	0.32	0.00	0.20	0.44	1.00
F0 x Intensity x Duration	-0.01	0.06	-0.18	0.854	-0.02	0.00	-0.15	0.10	0.36

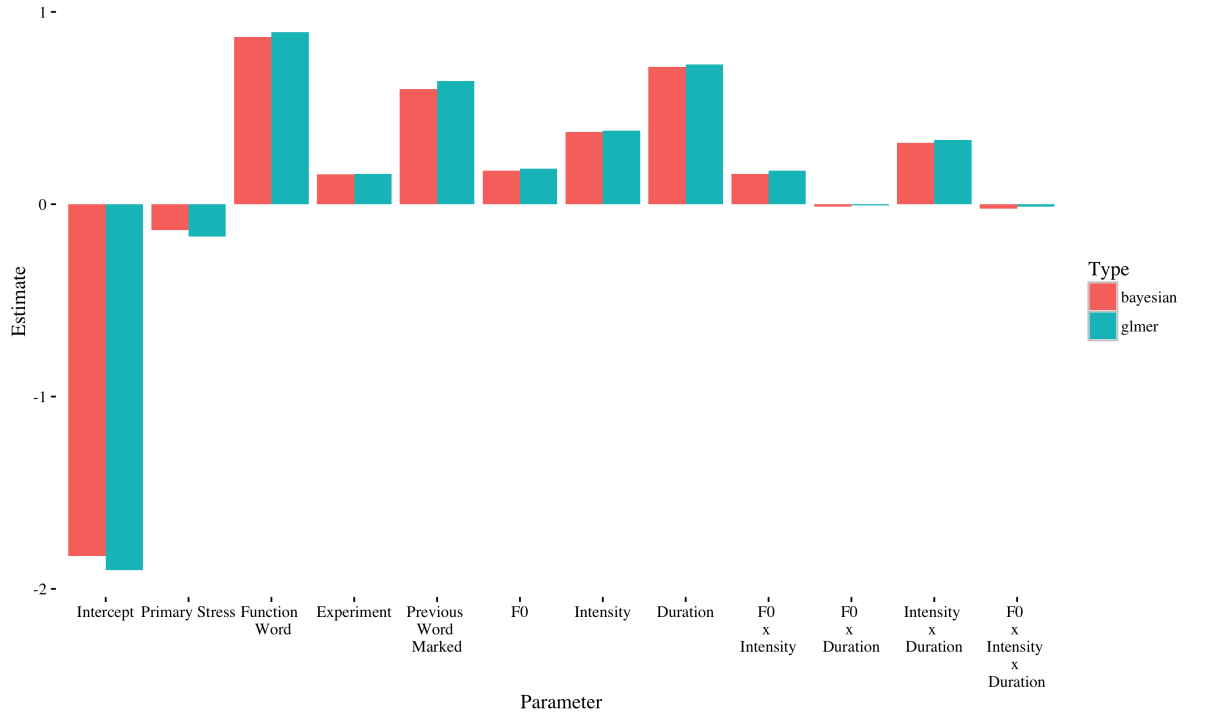


Figure 5: Plot of the beta estimates from the glmer and the Bayesian models for Primary Stress, Function Word, Experiment, Previous word Marked, and F0, Intensity, Duration and the two and three way interactions between them

The results for the standard mixed effects model fit with *lme4* and for the Bayesian model fit with *rstan* are summarized in Table 4. As is clear from the table, and also illustrated in Figure 5, the value of the estimates of the Bayesian implementation of the model hew very closely to the results of the glmer, in direction, size, and significance level. The only predictor with a change in interpretation is the Experimental Instructions, in that for the Bayesian model there is weak evidence for an effect– while the confidence interval contains 0, there is a 96% posterior probability that it has a real effect. This is not a predictor of theoretical interest, but it is crucial to the analysis of this work to

know whether data from these two different instructions can be pooled. This model provides a clear-cut example of how the Bayesian model may arrive at essentially the same results as the glmer. We include this dataset in part to show that the results of the Bayesian model will not differ widely from the glmer unless (as in the behavioral study) the data support that difference. Here, we see that the results of the glmer do not change when implemented with a model that includes random variance, although we see again that the glmer tends to overestimate effect sizes, while the Bayesian model is more conservative. While the two models achieve the same results, we can make more confident (and valid) conclusions from a Bayesian implementation in cases such as this one where the Bayesian model converges on the maximal random effects structures while the glmer does not.

4. Discussion

We have given two examples of models in which, if using only glmers in *lme4*, the analyst is forced to drop predictor and variance terms from the model, despite having strong *a priori* hypotheses that these terms will affect the dependent variable. Moreover, we underscore that unlike traditional model comparison, these terms are dropped not after testing to see what variance they explain, but rather left out in an arbitrary order in response to an error message.

Further motivation for the use of the Bayesian model is found by inspecting the variance components (random effects) estimated by the Bayesian model. The fact that a model run in *lme4* does not converge could be attributed to any of the convergence checks, including zero variance for one of the random effects. However, when a model does not converge one cannot inspect the random effects, because one can't be confident of the results of a model that did not converge— it is a statistical analyst's catch 22. Using a Bayesian model, not only is convergence more likely, the analyst can also inspect plots of the estimates of the variance for a given random effect, and confirm whether there are any random effects included in the model with zero or near-zero variance. If there

are terms with zero or low variance, the model shows this straightforwardly, and there is no need to re-run the model with these random effects left out.

Still, one might examine an instance such as the perception study reported here where the two methods yield essentially the same results and ask what is gained from a Bayesian implementation. We emphasize that even in this case, the Bayesian implementation is still preferred for several reasons. Firstly, because they are the results of a maximal model, the results of the Bayesian implementation of the model incorporate all expected variability. To put it another way: the analyst no longer need worry that (as in the behavioral task reported above) their results are spurious products of an unreliable model. Secondly, rather than running multiple iterations of glmers and searching for a model that achieves convergence, the analyst can begin with the original model that is desired based on their theoretical predictions, and run others as desired for model comparison. This allows for pre-registration of the analysis, a crucial step for increased reliability in the field overall. Furthermore, on a practical level the Bayesian technique saves time spent running multiple glmers (and bootstrapping p-values, and comparing models, etc.)

The current methods to deal with convergence errors outlined in section 1.2 all assume that the primary cause of convergence errors is the inclusion of a zero or near zero covariance parameter in the mixed effects model. This follows Bates et al (2015), who state *we explain that failure of convergence of model estimation is typically not a consequence of a suboptimal estimation algorithm, but rather an indicator of a model specification that is too complex to be properly supported by the data*. We have demonstrated in both data sets in this paper that convergence errors may come even with proper model specification and no inclusion of zero variance. The assessment of whether or not the estimating algorithm is *suboptimal* should best be determined by experts in computational statistics.

What is noticeably absent from literature advocating mixed effects models via *lme4* (e.g. Baayen et al., 2008; Barr et al., 2013; Gries, 2015) is reference to the technical statistical literature on mixed effects models. It is important

for researchers using mixed effects models to realize that they are still not fully theorized within the statistical literature¹⁷. From Hodges(2014:xxxiii-xxxiv):

A lot of academics think mixed linear models are completely understood, when in fact they are still largely not understood...the new methods [i.e. mixed effects models] of the last three decades are so complex that it may never be possible to prove theorems about them. We can, however, make progress by approaching our black-box methods in the same way our scientific colleagues approach nature's black-box methods, by prying them open gradually and indirectly if necessary.

When the statistical literature on mixed effects models is examined, it becomes clear that there are many open questions that still need to be solved for mixed effects models (e.g. Demidenko, 2013: xxiii-xxvii). Convergence failure can be due to other issues besides true zero or small variance components included in the model. This is true across software packages: Hodges (2014: 303-412) uses PROC MIXED in SAS, and presents multiple mixed effects model results from real data sets that are, as he labels them, *mysterious, inconvenient or wrong*. So researchers can be sure that this is not an *R* problem and that if they just used SAS, that would overcome this. In fact, it is even more difficult to determine what convergence tests SPSS & SAS actually do, because of their proprietary nature unlike with *lme4* where researchers can see the actual implementation.

Fully specified Bayesian models allow for the possibility of a small value for subject or item variance or even 0 variance, meaning that researchers can get useful, statistically valid results even when convergence is due to near zero variance.

¹⁷For example, for both linear models and generalized linear models there are mathematical results that allow you to simultaneously assess all parameters in the model whereas for mixed effects models there are no agreed on mechanism to assess an optimal random effects structure present in the literature.

5. Conclusion

In summary, we consider mixed effect models implemented in a frequentist framework (as in `glmer()` in *lme4*) to have three main advantages: they model both fixed and random effects; they are well-documented and accessible to beginner users; they are widely used. However, these advantages are undercut by the problems introduced by convergence errors. Mixed effect models implemented in a Bayesian framework retain all three advantages that we have identified while decreasing the chance of convergence errors.

Overall, we have made three central arguments in this paper:

1. Convergence errors are a common problem, but may not indicate that a model is poorly specified.
2. Current methods of addressing convergence errors do not always follow good statistical practice.
3. A fully specified Bayesian implementation of mixed effect models is one way to avoid these errors.

We acknowledge that a switch to Bayesian methods comes at a cost: researchers, reviewers, and readers alike will have to learn to program and/or interpret a new type of model. We recognize that many researchers have expressed annoyance and even anger as newer complex models are advocated in the language sciences, and that there is a palpable frustration among some scholars that these newer complex models are in fact being used merely to function as a gatekeeping device for those who do not speak the language of programming and statistics to be excluded from publication.

The purpose of this paper is not to insist that all researchers need to implement these even more complex models, but to foreground three items in the current intense debate around statistical models in the language sciences. First, the full Bayesian extension of the mixed effects model gives the researcher the ability to account for the inter-subject and inter-item variability, as in the unconstrained mixed-effects models; but, for the data presented the former produces results while the latter do not.

Second, and more crucially, the data and analyses in this paper have demonstrated that a lack of convergence does not automatically indicate a poorly specified model, and that the use of a fully specified Bayesian model can provide a more statistically valid set of results when the mixed effects estimation algorithm goes rogue. A fully specified Bayesian models allows researchers a greater chance of implementing a pre-registered and more accurate random effects structure rather than having to hope that the most appropriate model convergences for their data.

Third, as commonly used statistical methods become more complex, the need for transparency, and pre-registration grows. If authors are to present their studies as replicable instances of language science, documentation of the precise methods used is crucial, as is seen by the widely reported so-called “replicability crisis” in Psychology (Open Science Collaboration, 2015; Ioannidis, 2005). Current methods to address convergence errors do not allow for pre-registration because an author cannot pre-register an analysis by stating “I plan to run Model X and conduct model comparison, but convergence errors may force me to run any subset of my maximal model; I will continue to run models until one converges, and then conduct model comparison with any other models that happen to converge.” Running multiple models in search of convergence is doubly problematic because running many models is tantamount to running many statistical tests and waiting for optimal results. Fortunately, the Bayesian model offers a solution to this problem by allowing the researcher to run models and conduct model comparison with a much higher likelihood of useable results, even with zero or low-variance predictors remaining in the model.

6. Acknowledgements

AK received financial support from an Illinois Distinguished Fellowship from the University of Illinois. KS received financial support from a Doctoral Fellowship from the Social Sciences and Humanities Research Council of Canada, and from an Illinois Distinguished Fellowship from the University of Illinois.

Data collection was supported in part by NSF BCS-1251343 to Jennifer Cole and Jose I. Hualde, by NSF BCS-1349110 and BCS-1431324 to Darren Tanner, and by a grant from the Illinois Campus Research Board [grant number RB14158] to Darren Tanner. This research was supported by equipment funded from the Office of the Vice-Chancellor of Research at the University of Illinois at Urbana-Champaign to JR. Finally, thanks to Darren Tanner and Jennifer Cole for making their data available.

References

- Abdel Rahman, R., & Sommer, W. (2003). Does phonological encoding in speech production always follow the retrieval of semantic knowledge?: Electrophysiological evidence for parallel processing. *Cognitive Brain Research*, *16*, 372–382.
- Abdel Rahman, R., Van Turenout, M., & Levelt, W. J. (2003). Phonological encoding is not contingent on semantic feature retrieval: an electrophysiological study on object naming. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *29*, 850.
- Akhutina, T., Kurgansky, A., Polinsky, M., & Bates, E. (1999). Processing of grammatical gender in a three-gender system: Experimental evidence from russian. *Journal of Psycholinguistic Research*, *28*, 695–713.
- Alario, F.-X., Ferrand, L., Laganaro, M., New, B., Frauenfelder, U. H., & Segui, J. (2004). Predictors of picture naming speed. *Behavior Research Methods, Instruments, & Computers*, *36*, 140–155.
- Baayen, H. (2008). *Analyzing Linguistic Data*. Cambridge University Press.
- Baayen, R. H., Davidson, D. J., & Bates, D. M. (2008). Mixed-effects modeling with crossed random effects for subjects and items. *Journal of Memory and Language*, *59*, 390–412.

- Barr, D. J. (2013). Random effects structure for testing interactions in linear mixed-effects models. *Frontiers in psychology*, *4*, 328.
- Barr, D. J., Levy, R., Scheepers, C., & Tily, H. J. (2013a). Random effects structure for confirmatory hypothesis testing: Keep it maximal. *Journal of memory and language*, *68*, 255–278.
- Barr, D. J., Levy, R., Scheepers, C., & Tily, H. J. (2013b). Random effects structure for confirmatory hypothesis testing: Keep it maximal. *Journal of Memory and Language*, *68*, 255–278.
- Bates, D. M., Kliegl, R., Vasishth, S., & Baayen, H. (forthcoming). Parsimonious mixed models. *Journal of Memory and Language*, .
- Bates, E., Devescovi, A., Hernandez, A., & Pizzamiglio, L. (1996). Gender priming in italian. *Perception & Psychophysics*, *58*, 992–1004.
- Bishop, J. (2012). Information structural expectations in the perception of prosodic prominence. In P. Elordieta, G; Prieto (Ed.), *Prosody and Meaning*. Mouton De Gruyter.
- Breen, M., & Clifton Jr., C. (2011). Stress matters: Effects of anticipated lexical stress on silent reading. *Journal of Memory and Language*, *64*, 153–170.
- Breen, M., Fedorenko, E., Wagner, M., & Gibston, E. (2010). Acoustic correlates of information structure. *Language and Cognitive Processes*, *25*, 1044–1098.
- Carpenter, B., Gelman, A., Hoffman, M., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M. A., Guo, J., Li, P., & Riddell, A. (2016). Stan: A probabilistic programming language. *J Stat Softw*, .
- Cole, J., Mo, Y., & Baek, S. (2010a). The role of syntactic structure in guiding prosody perception with ordinary listeners and everyday speech. *Language and Cognitive Processes*, *25*, 1141–1177.

- Cole, J., Mo, Y., & Hasegawa-Johnson, M. (2010b). Signal-based and expectation-based factors in the perception of prosodic prominence. *Laboratory Phonology*, (pp. 425–452).
- Demidenko, E. (2013). Mixed models: theory and applications with R.
- Fine, A. B., Jaeger, T. F., Farmer, T. A., & Qian, T. (2013). Rapid expectation adaptation during syntactic comprehension. *PloS one*, 8, e77661.
- Flaxman, S., Gelman, A., Neill, D., Smola, A., Vehtari, A., & Wilson, A. G. (2015). Fast hierarchical gaussian processes. *Manuscript in preparation*, .
- Gelman, A., Carlin, J., Stern, H., Dunson, D., Vehtari, A., & Rubin, D. (2014a). *Bayesian Data Analysis*. Taylor and Francis.
- Gelman, A., Carlin, J. B., Stern, H. S., & Rubin, D. B. (2014b). *Bayesian data analysis* volume 2. Chapman & Hall/CRC Boca Raton, FL, USA.
- Gelman, A., Jakulin, A., Pittau, M. G., & Su, Y.-S. (2008). A weakly informative default prior distribution for logistic and other regression models. *The Annals of Applied Statistics*, (pp. 1360–1383).
- Gomez, P., Ratcliff, R., & Perea, M. (2007). A model of the go/no-go task. *Journal of Experimental Psychology: General*, 136, 389.
- Grabe, E., Warren, P., & Warren, P. (1995). Stress Shift: do speakers do it or do listeners hear it? In B. Connell, & A. Arvaniti (Eds.), *Phonology and Phonetic Evidence: Papers in Laboratory Phonology IV* (pp. 95–110). Cambridge University Press.
- Gries, S. T. (2015). The most under-used statistical method in corpus linguistics: multi-level (and mixed-effects) models. *dx.doi.org*, 10, 95–125.
- Hardin, J. W., & Hilbe, J. M. (2007). *Generalized linear models and extensions*. Stata press.

- Imrey, P. B., Koch, G. G., & Stokes, M. E. (1981). Categorical data analysis: some reflections on the log linear model and logistic regression. part i: historical and methodological overview. *International Statistical Review/Revue Internationale de Statistique*, (pp. 265–283).
- Ioannidis, J. (2005). Why most published research findings are false. *PLoS Med*, 2.
- Jaeger, T. F. (2009). Random effect: Should i stay or should i go? URL: <https://hlplab.wordpress.com/2009/05/14/random-effect-structure/>.
- Jescheniak, J. D., & Levelt, W. J. (1994). Word frequency effects in speech production: Retrieval of syntactic information and of phonological form. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 20, 824.
- Johnson, D. E. (2009). Getting off the goldvarb standard: Introducing rbrul for mixed-effects variable rule analysis. *Language and Linguistic Compass*, 3, 359–383.
- Kimball, A. E., & Cole, J. (2014). Avoidance of Stress Clash in Perception of Conversational American English. *Proceedings of Speech Prosody*, 7.
- Kimball, A. E., Yiu, L. K., & Watson, D. (2016). Word recall is affected by surrounding metrical context. *unpublished*, .
- Kimball, J., A E; Cole (2016). Perception of regular meter in conversational american english. *unpublished*, .
- Kruschke, J. (2013). Bayesian estimation supersedes the t test. *Journal of Experimental Psychology: General*, 142, 573–603.
- Lange, K. (2010). *Numerical analysis for statisticians*. Springer Science & Business Media.
- Levitt, A. G., & Healy, A. F. (1985). The roles of phoneme frequency, similarity, and availability in the experimental elicitation of speech errors. *Journal of Memory and Language*, 24, 717–733.

- Luke, S. G. (2016). Evaluating significance in linear mixed-effects models in r. *Behavior Research Methods*, (pp. 1–9).
- Matuschek, H., Kliegl, R., Vasishth, S., Baayen, H., & Bates, D. (2015). Balancing type i error and power in linear mixed models. *arXiv preprint arXiv:1511.01864*, .
- McCullagh, P., & Nelder, J. (1983). *Generalized linear models*. Chapman & Hall.
- McElreath, R. (2016). *Statistical Rethinking: A Bayesian Course with Examples in R and STAN*. CRC Press.
- Molenberghs, G., & Verbeke, G. (2005). *Models for Discrete Longitudinal Data*. New York: Springer Science+ Business Media.
- Monahan, J. F. (2011). *Numerical methods of statistics*. Cambridge University Press.
- Nicenboim, B., & Vasishth, S. (2016). Statistical methods for linguistic research: Foundational ideas—part ii. *Language and Linguistics Compass*, *10*, 591–613.
- Open Science Collaboration (2015). Estimating the reproducibility of psychological science. *Science*, *349*, aac4716–aac4716.
- Opitz, A., & Pechmann, T. (2016). Gender features in german: Evidence for underspecification. *The Mental Lexicon*, *11*, 216–241.
- Pitt, M. A., Dilley, L., Kiesling, S., Raymond, W., Hume, E., & Fosler-Lussier, E. (2007). Buckeye Corpus of Conversational Speech [second release].
- Quené, H., & Port, R. F. (2005). Effects of timing regularity and metrical expectancy on spoken-word perception. *Phonetica*, *62*, 1–13.
- Schielzeth, H., & Forstmeier, W. (2009). Conclusions beyond support: overconfident estimates in mixed models. *Behavioral Ecology*, *20*, 416–420.

- Schmitt, B. M., Münte, T. F., & Kutas, M. (2000). Electrophysiological estimates of the time course of semantic and phonological encoding during implicit picture naming. *Psychophysiology*, *37*, 473–484.
- Selkirk, E. (1995). The Prosodic Structure of Function Words. In J. L. Morgan, & K. Demuth (Eds.), *Signal to Syntax Bootstrapping from Speech to Grammar in Early Acquisition*.
- Shantz, K., & Tanner, D. (2016). Talking out of order: task order and retrieval of grammatical gender and phonology in lexical access. *Language, Cognition and Neuroscience*, *0*, 1–22. doi:10.1080/23273798.2016.1221510.
- Strijkers, K., Costa, A., & Thierry, G. (2009). Tracking lexical access in speech production: electrophysiological correlates of word frequency and cognate effects. *Cerebral cortex*, (p. bhp153).
- Strijkers, K., Holcomb, P. J., & Costa, A. (2011). Conscious intention to speak proactively facilitates lexical access during overt object naming. *Journal of memory and language*, *65*, 345–362.
- Th. Gries, S. (2015). The most under-used statistical method in corpus linguistics: multi-level (and mixed-effects) models. *Corpora*, *10*, 95–125.
- Tilsen, S. (2011). Metrical regularity facilitates speech planning and production. *Laboratory Phonology*, *2*.
- Van Turennout, M., Hagoort, P., & Brown, C. M. (1997). Electrophysiological evidence on the time course of semantic and phonological processes in speech production. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *23*, 787.
- Van Turennout, M., Hagoort, P., & Brown, C. M. (1998). Brain Activity During Speaking: From Syntax to Phonology in 40 Milliseconds. *Science*, *280*, 572–574.

Vogel, I., Bunnell, T., & Hoskins, S. (1995). The Phonology and Phonetics of the Rhythm Rule. In B. Connell, & A. Arvaniti (Eds.), *Phonology and Phonetic Evidence: Papers in Laboratory Phonology IV* (pp. 111–127). Cambridge University Press.

Zheng, X., & Pierrehumbert, J. B. (2010). The effects of prosodic prominence and serial position on duration perception. *The Journal of the Acoustical Society of America*, 128, 851–859.

7. Appendix A: Stan Code to Generate Models

All the code to generate results has been posted online at Github and reproduced fully below and in Appendix C.

```
// Stan model for mixed effects logistic regression
// with random effects for subject and item

// the code is written assuming that both subjects and items
// have at least one slope. It is not written for models with
// only one grouping factor or intercept-only models. These
// would require a few simple modifications. See the online
// supplement for sample R code.

// the model matrix x should have columns with the following
// order and then be converted to CSR format:
//   fixed effects
//   subject effects ordered by subject
//   within-subject, the effects should be in the same order
//   item effects ordered by item
//   within-item, the effects should be in the same order

// the binary response y should be passed as integer 0/1

// the (reparameterized) priors are:
//   fixed effects
//   beta ~ student_t(5,0,sigma_beta)
//   sigma_beta ~ half_normal(0,1)
```

```

// random effects
// sigma_group ~ half_normal(0,1)
// omega_group ~ lkj_corr(2)
// gamma_group ~ mult_normal(0,Sigma_group)
// where Sigma_group = diag(sigma_group)
// * omega_group * t(diag(sigma_group))

functions {
  // this function takes a vector and turns it into
  // a matrix with R rows and C columns, loading
  // the values by row
  matrix vec_to_matrix_by_row(vector v, int R, int C){
    matrix [R,C] m;
    for (r in 1:R) m[r] = v[(C*(r-1)+1):(C*r)]';
    return m;
  }
}

data {
  int<lower=2> N; // number of observations
  int<lower=2> S; // number of subjects
  int<lower=2> I; // number of items

  int<lower=1> P; // number of fixed effects
  int<lower=1,upper=P> QS; // number of subject effects
  int<lower=1,upper=P> QI; // number of item effects

  int<lower=0,upper=1> y[N]; // binary response

  // sparse model matrix (CSR)
  int<lower=1> nz; // number of non-zero elements in x
  vector [nz] x_w; // non-zero elements in x
  int x_v[nz]; // column indices for x_w
  int x_u[N+1]; // row-start indices for x
}

transformed data {
  int K; // number of columns in x

```

```

int SF; // first subject effect column in x
int SL; // last subject effect column in x
int IF; // first item effect column in x
int IL; // last item effect column in x

K = P + S * QS + I * QI;
SF = P + 1;
SL = P + S * QS;
IF = SL + 1;
IL = SL + I * QI;
}

parameters {
  vector[P] beta_raw;
  vector<lower=0>[P] tau_beta;
  real<lower=0> sigma_beta;

  matrix[QS,S] gamma_subj_raw;
  vector<lower=0>[QS] sigma_subj; // subject effect SDs
  cholesky_factor_corr[QS] omega_subj_raw;

  matrix[QI,I] gamma_item_raw;
  vector<lower=0>[QI] sigma_item; // item effect SDs
  cholesky_factor_corr[QI] omega_item_raw;
}

transformed parameters {
  vector[K] coef; // all coefficients
  vector[N] y_hat; // predicted log-odds

  // transform fixed effects
  for(p in 1:P)
    coef[p] = beta_raw[p] * sigma_beta / sqrt(tau_beta[p]);

  // transform subject effects
  coef[SF:SL]
    = to_vector(rep_matrix(sigma_subj,S)
      .* (omega_subj_raw * gamma_subj_raw));

```

```

// transform item effects
coef[IF:IL]
    = to_vector(rep_matrix(sigma_item, I)
        .* (omega_item_raw * gamma_item_raw));

// y-hat = x * coef
y_hat = csr_matrix_times_vector(N, K, x_w, x_v, x_u, coef);
}

model {
    beta_raw ~ normal(0, 1);
    tau_beta ~ gamma(2.5, 2.5);
    sigma_beta ~ normal(0, 1);

    to_vector(gamma_subj_raw) ~ normal(0, 1);
    sigma_subj ~ normal(0, 1);
    omega_subj_raw ~ lkj_corr_cholesky(2);

    to_vector(gamma_item_raw) ~ normal(0, 1);
    sigma_item ~ normal(0, 1);
    omega_item_raw ~ lkj_corr_cholesky(2);

    y ~ bernoulli_logit(y_hat); // logistic model defined
}

generated quantities {
    vector[P] beta; // fixed effects
    matrix[S, QS] gamma_subj; // subject effects
    matrix[I, QI] gamma_item; // item effects
    matrix[QS, QS] omega_subj; // correlation in subject effects
    matrix[QI, QI] omega_item; // correlation in item effects

    beta = coef[1:P];
    gamma_subj = vec_to_matrix_by_row(coef[SF:SL], S, QS);
    gamma_item = vec_to_matrix_by_row(coef[IF:IL], I, QI);
    omega_subj = tcrossprod(omega_subj_raw);
    omega_item = tcrossprod(omega_item_raw);
}

```


}

8. Appendix B: Graphs of Random Variance

8.1. Behavioral task

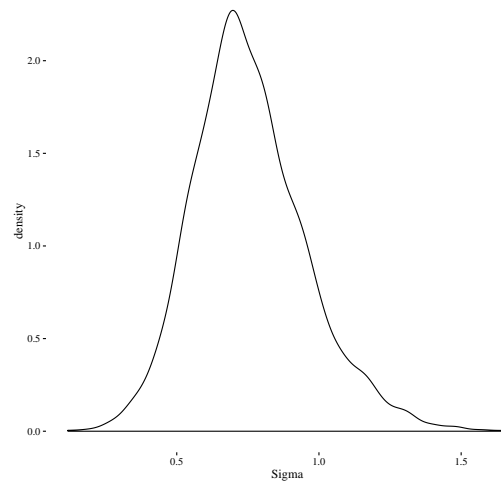


Figure 6: Density plot of the variance estimates for subject intercept produced by the Bayesian model

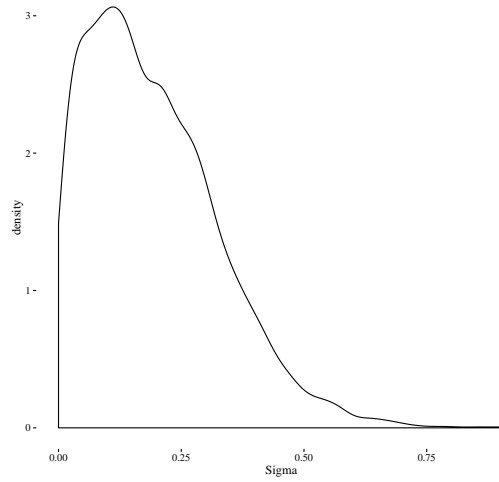


Figure 7: Density plot of the variance estimates for the random slope of trial by subjects produced by the Bayesian model

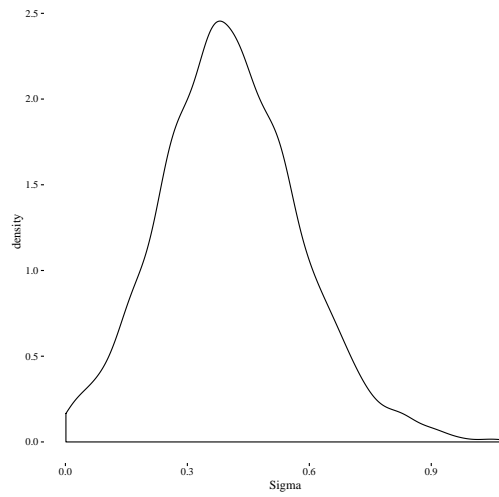


Figure 8: Density plot of the variance estimates for the random slope of condition by subjects produced by the Bayesian model

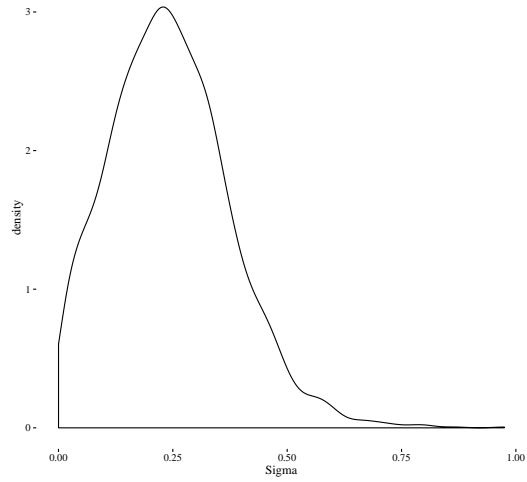


Figure 9: Density plot of the variance estimates for the random slope of gender by subjects produced by the Bayesian model

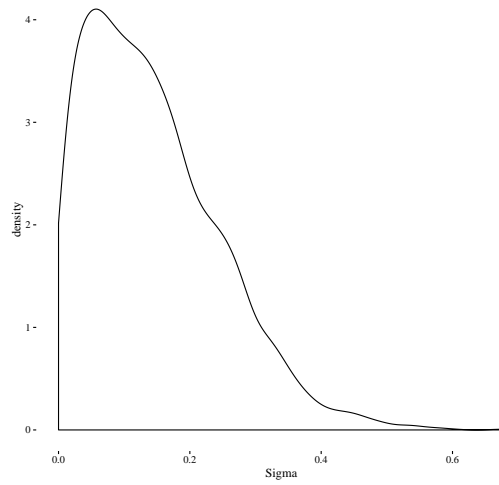


Figure 10: Density plot of the variance estimates for the random slope of initial sound by subjects produced by the Bayesian model

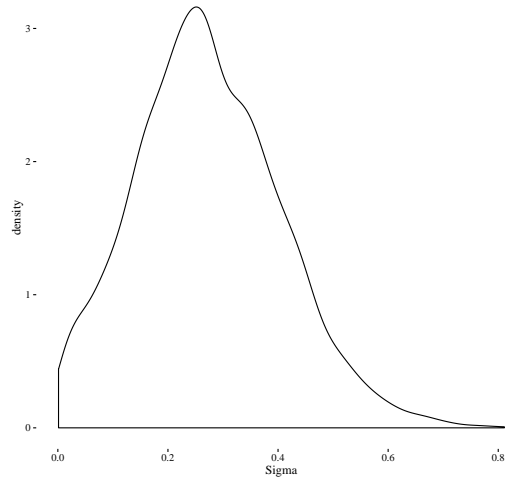


Figure 11: Density plot of the variance estimates for the random slope of frequency by subjects produced by the Bayesian model

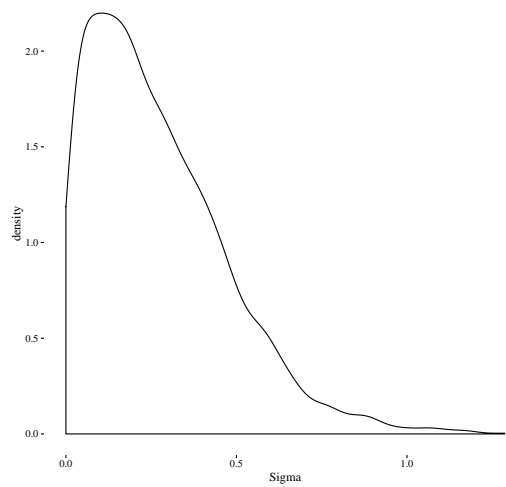


Figure 12: Density plot of the variance estimates for the random slope of syllables (linear) by subjects produced by the Bayesian model

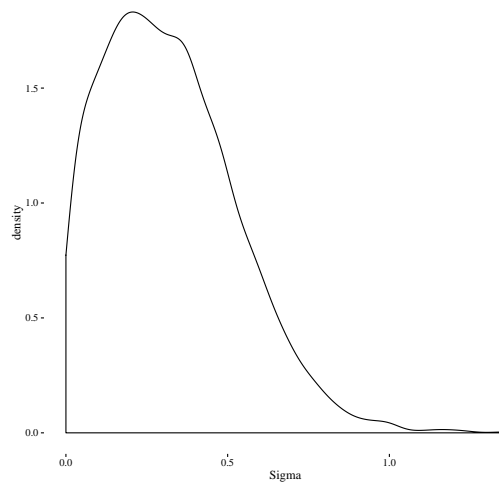


Figure 13: Density plot of the variance estimates for the random slope of syllables (quadratic) by subjects produced by the Bayesian model

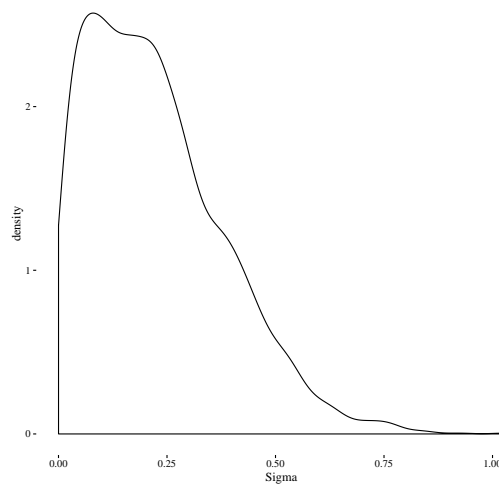


Figure 14: Density plot of the variance estimates for the random slope of the interaction of trial and condition by subjects produced by the Bayesian model

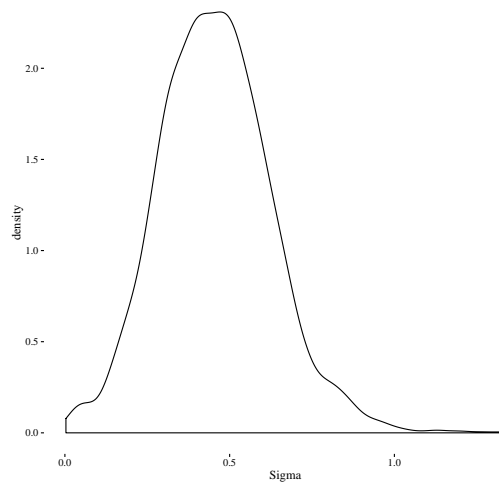


Figure 15: Density plot of the variance estimates for the random slope of the interaction of trial type and condition produced by the Bayesian model

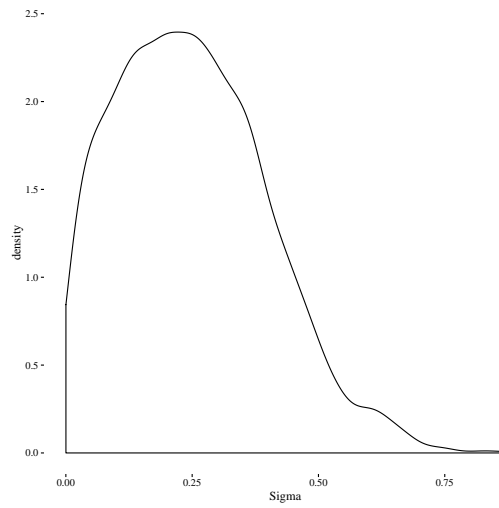


Figure 16: Density plot of the variance estimates for item intercept produced by the Bayesian model

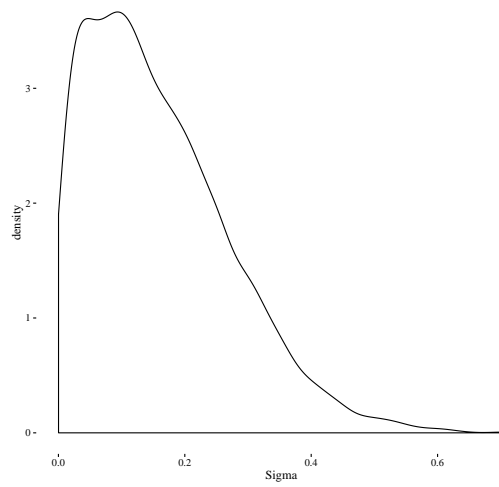


Figure 17: Density plot of the variance estimates for the random slope of trial by items produced by the Bayesian model

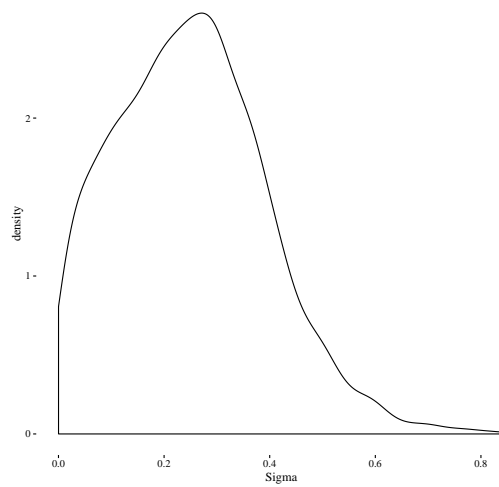


Figure 18: Density plot of the variance estimates for the random slope of trial type by items produced by the Bayesian model

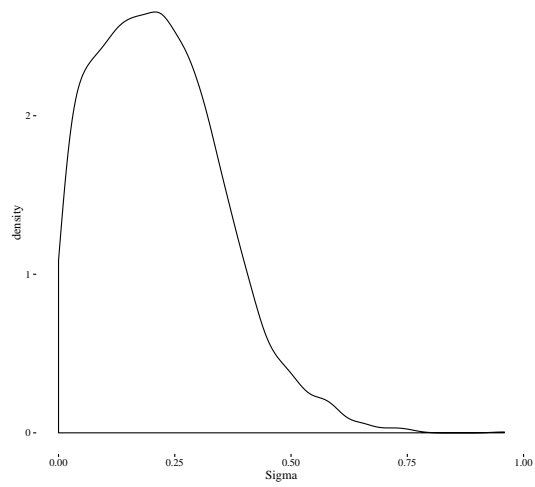


Figure 19: Density plot of the variance estimates for the random slope of condition by items produced by the Bayesian model

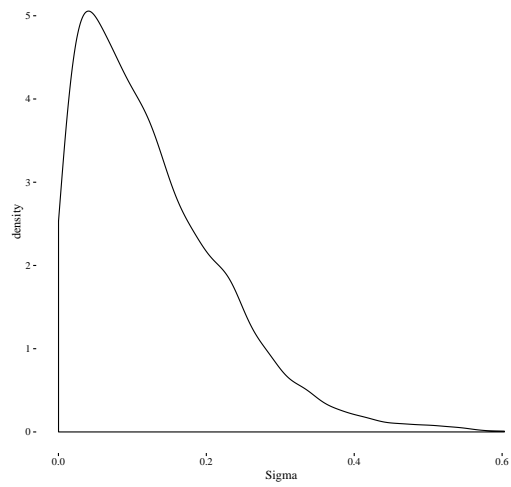


Figure 20: Density plot of the variance estimates for the random slope of task order by items produced by the Bayesian model

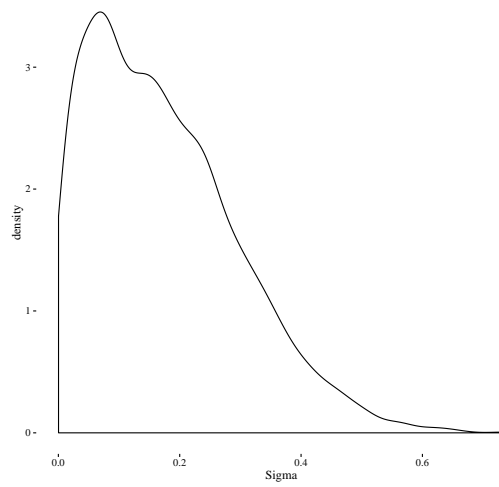


Figure 21: Density plot of the variance estimates for the random slope of the interaction of trial and task order by items produced by the Bayesian model

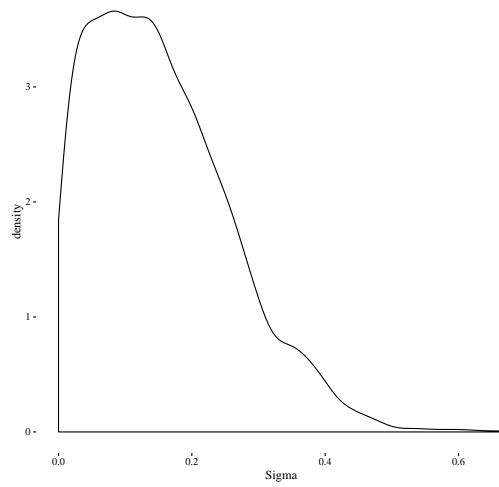


Figure 22: Density plot of the variance estimates for the random slope of the interaction of trial type and task order by items produced by the Bayesian model

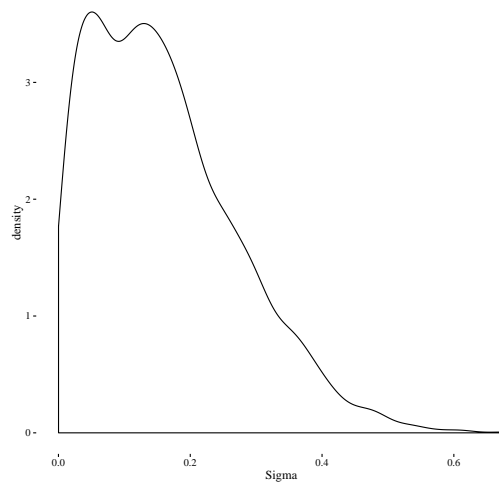


Figure 23: Density plot of the variance estimates for the random slope of the interaction of trial and condition by items produced by the Bayesian model

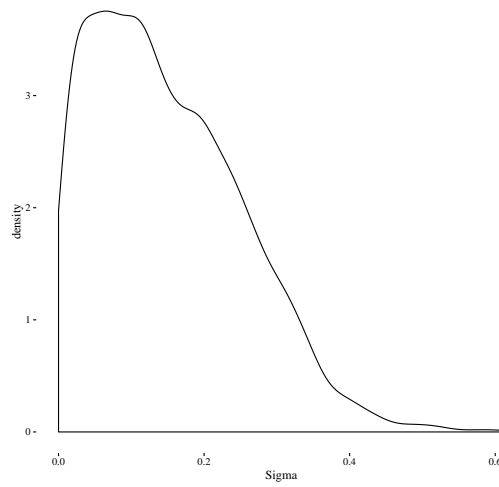


Figure 24: Density plot of the variance estimates for the random slope of the interaction of trial type and condition by items produced by the Bayesian model

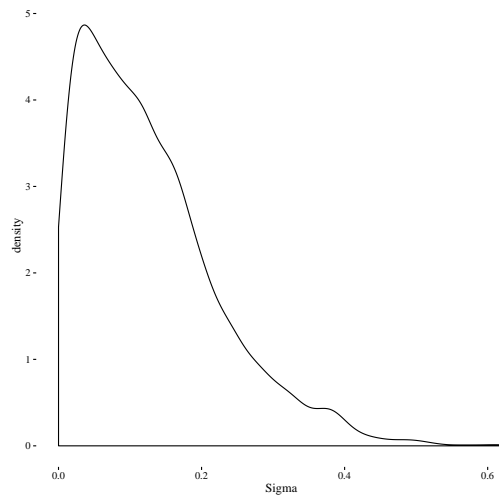


Figure 25: Density plot of the variance estimates for the random slope of the interaction of task order and condition by items produced by the Bayesian model

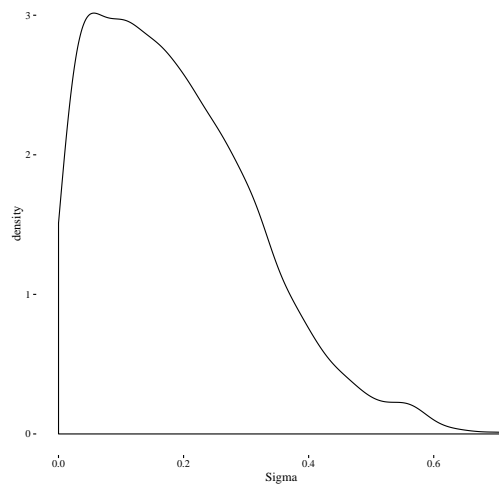


Figure 26: Density plot of the variance estimates for the random slope of the interaction of trial, task order and condition by items produced by the Bayesian model

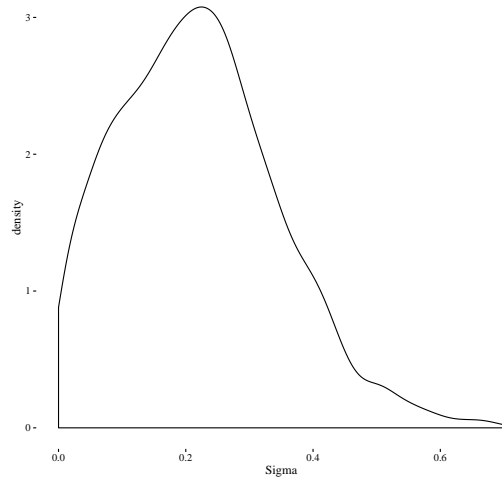


Figure 27: Density plot of the variance estimates for the random slope of the interaction of trial type, task order and condition by items produced by the Bayesian model

8.2. Perception Study

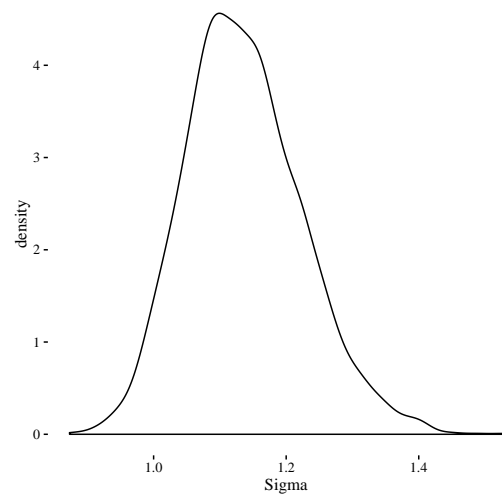


Figure 28: Density plot of the variance estimates for subject intercept produced by the Bayesian model

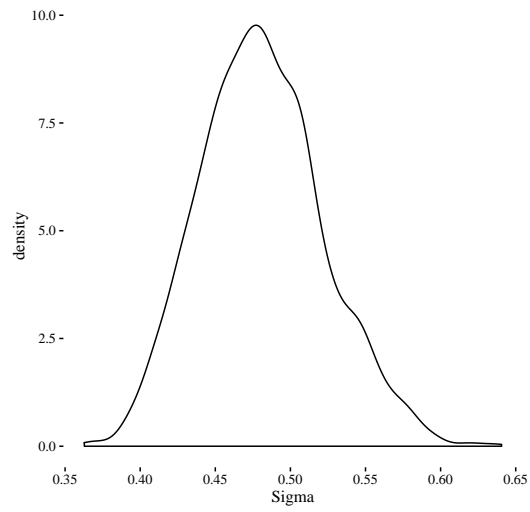


Figure 29: Density plot of the variance estimates for the random slope of function word by subjects produced by the Bayesian model

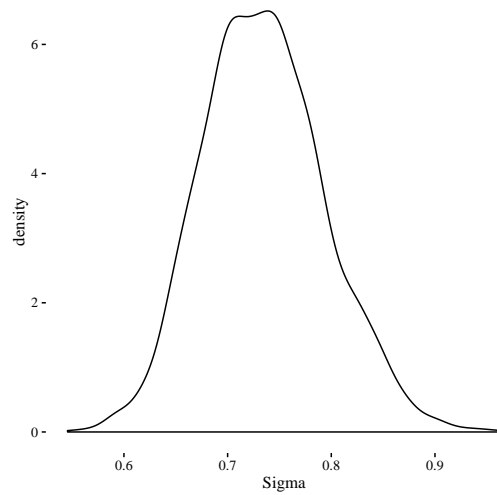


Figure 30: Density plot of the variance estimates for the random slope of primary stress by subjects produced by the Bayesian model

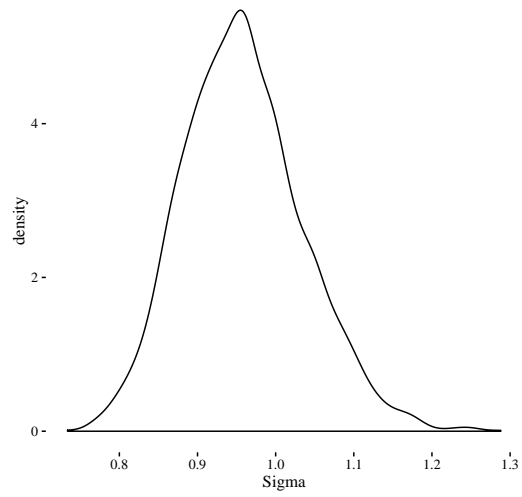


Figure 31: Density plot of the variance estimates for the random slope of previous word marked by subjects produced by the Bayesian model

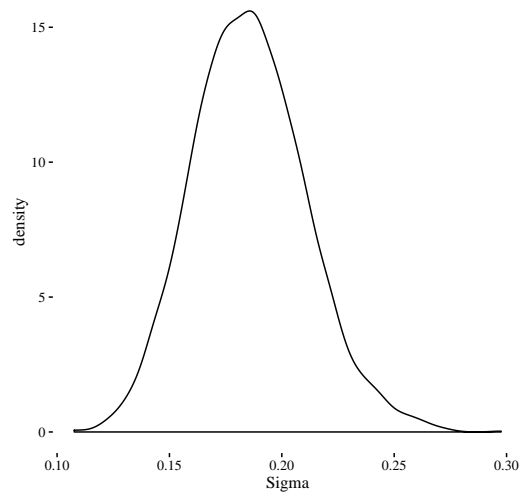


Figure 32: Density plot of the variance estimates for the random slope of F0 by subjects produced by the Bayesian model

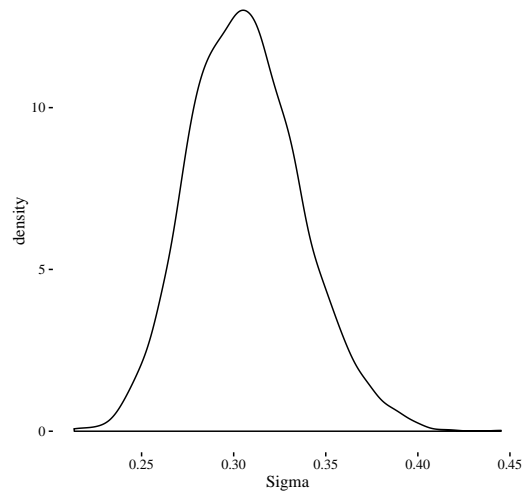


Figure 33: Density plot of the variance estimates for the random slope of intensity by subjects produced by the Bayesian model

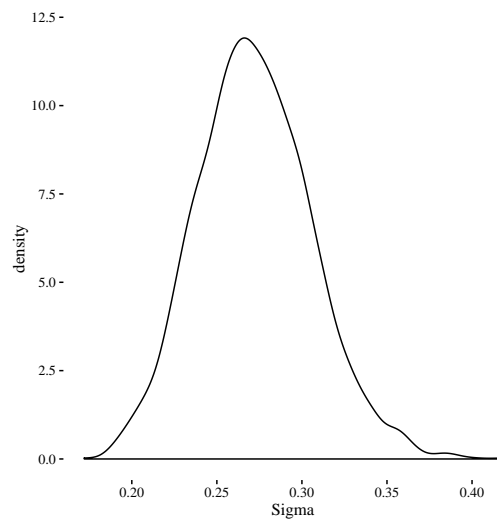


Figure 34: Density plot of the variance estimates for the random slope of duration by subjects produced by the Bayesian model

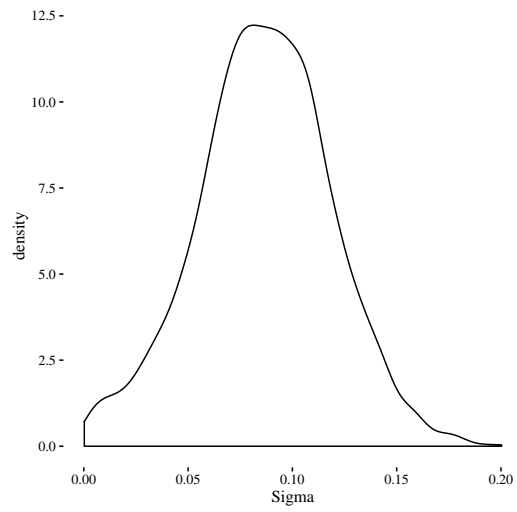


Figure 35: Density plot of the variance estimates for the random slope of the interaction of f0 and duration by subjects produced by the Bayesian model

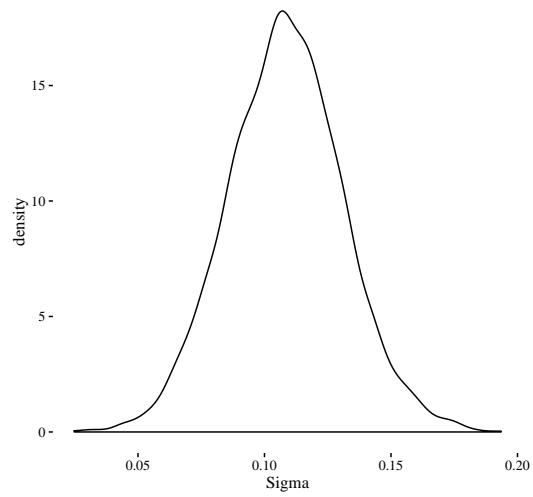


Figure 36: Density plot of the variance estimates for the random slope of the interaction of f0 and intensity by subjects produced by the Bayesian model

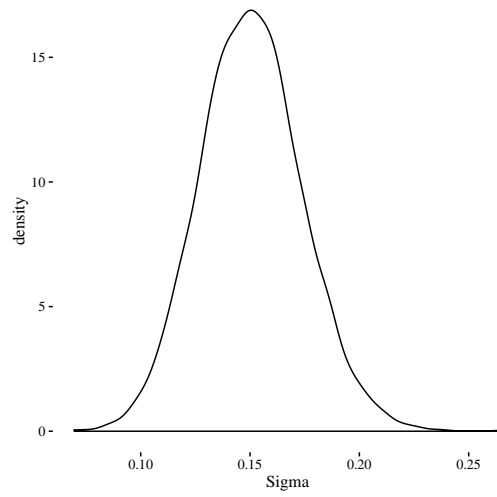


Figure 37: Density plot of the variance estimates for the random slope of the interaction of intensity and duration by subjects produced by the Bayesian model

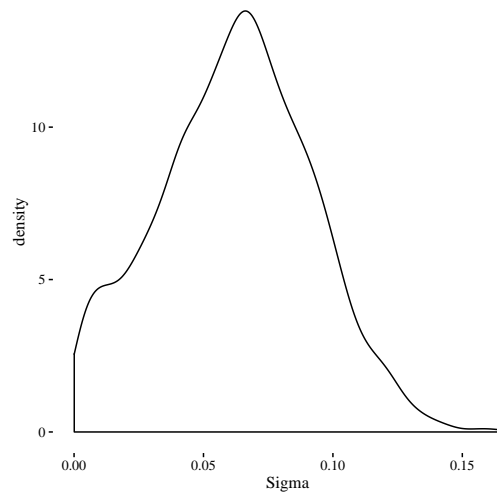


Figure 38: Density plot of the variance estimates for the random slope of the interaction of f0, intensity and duration by subjects produced by the Bayesian model

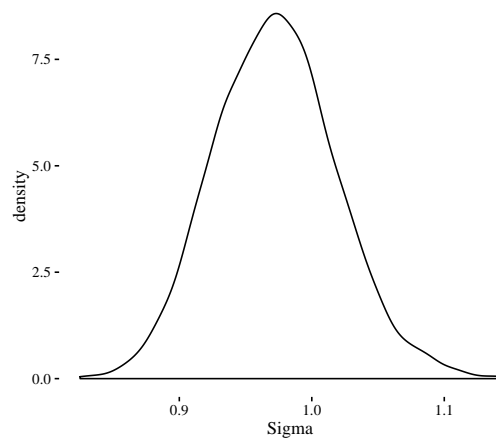


Figure 39: Density plot of the variance estimates for item intercept produced by the Bayesian model

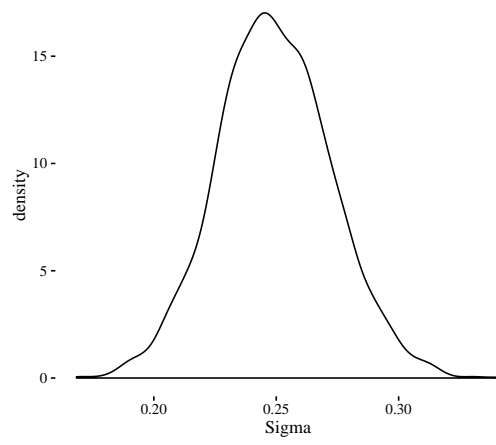


Figure 40: Density plot of the variance estimates for the random slope of experiment by items produced by the Bayesian model

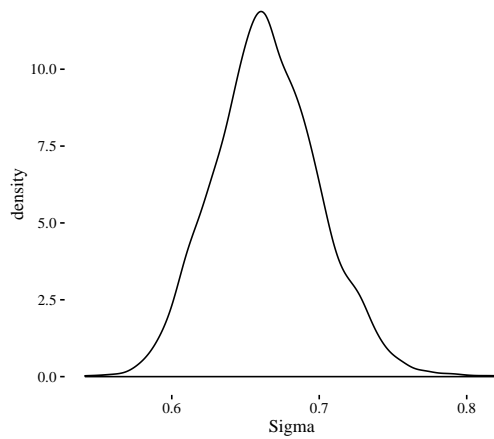


Figure 41: Density plot of the variance estimates for the random slope of previous word marked by items produced by the Bayesian model

9. Appendix C: R Code for all lme4 and RStan models

9.1. Behavioral Task

```
library(lme4)
library(rstan)
rstan_options(auto_write = TRUE)
options(mc.cores = parallel::detectCores())
options(contrasts=c("contr.sum", "contr.poly"))
source("getStanDataInfo.R")
source("getSummary.R")
source("printSummary.R")

behavioral_data <- read.csv("behavioral_task_data.csv", header=T)
behavioral_data$Syllables <- factor(behavioral_data$Syllables,
  ordered=T)
behavioral_data$Trial <- scale(behavioral_data$Trial)[,1]
behavioral_data$SUBTLEX_LogFrequency <- scale(behavioral_data$
  SUBTLEX_LogFrequency)[,1]

max_form <- Accuracy ~ (Trial + GoNoGo) * GoNoGo_Group *
  HandDecision +
  Gender + InitialSound + SUBTLEX_LogFrequency + Syllables +
```

```

(1 + Trial * HandDecision + GoNoGo * HandDecision + Gender +
InitialSound + SUBTLEX_LogFrequency + Syllables | Subject) +
(1 + (Trial + GoNoGo) * GoNoGo_Group * HandDecision | Item)

##### lme4
Full_Model_lme4 <- glmer(formula = max_form, data = behavioral_data
, family = binomial)
# Model failed to converge: degenerate Hessian with 2 negative
eigenvalues
# Time: 5.75 hours

Full_Model_bobyqa <- glmer(
formula = max_form, data = behavioral_data, family = binomial,
glmerControl(optimizer="bobyqa"))
# Model failed to converge: degenerate Hessian with 1 negative
eigenvalues

Full_Model_Nelder_Mead <- glmer(
formula = max_form, data = behavioral_data, family = binomial,
glmerControl(optimizer="Nelder_Mead"))
# Model failed to converge: degenerate Hessian with 15 negative
eigenvalues

Intercepts_Only <- glmer(
Accuracy ~ (Trial + GoNoGo) * GoNoGo_Group * HandDecision +
Gender + InitialSound + SUBTLEX_LogFrequency + Syllables +
(1|Subject) + (1|Item),
family = binomial, data = behavioral_data)
# Model failed to converge with max|grad| = 0.00426312 (tol =
0.001, component 1)

One_Intercept<- glmer(
Accuracy ~ (Trial + GoNoGo) * GoNoGo_Group * HandDecision +
Gender + InitialSound + SUBTLEX_LogFrequency + Syllables + (1|
Subject),
family = binomial, data = behavioral_data)
# Converges

```

```
##### stan
di <- getStanDataInfo(formula = max_form, data = behavioral_data,
  subj = "Subject", item = "Item")
Full_Model_stan <- stan(file = "glmer_logistic.stan", data = di$
  data,
  chains = 3, iter = 2000, warmup = 1000, refresh = 100,
  pars = di$info$keep, open_progress = TRUE,
  sample_file = "behavioral_task.stancsv")
Full_Model_stan_summ <- getSummary(Full_Model_stan, di)
printSummary(Full_Model_stan_summ)
# divergent transitions
# Time: 1.5 hours

Full_Model_stan <- stan(file = "glmer_logistic.stan", data = di$
  data,
  chains = 3, iter = 2000, warmup = 1000, refresh = 100,
  pars = di$info$keep, open_progress = TRUE, control = list(adapt_
    delta = 0.99),
  sample_file = "behavioral_task.stancsv")
Full_Model_stan_summ <- getSummary(Full_Model_stan, di)
printSummary(Full_Model_stan_summ)
# Converges
# Time: 3 hours
```

9.2. Perception Study

```
library(lme4)
library(rstan)
rstan_options(auto_write = TRUE)
options(mc.cores = parallel::detectCores())
options(contrasts=c("contr.sum", "contr.poly"))
source("getStanDataInfo.R")
source("getSummary.R")
source("printSummary.R")

perception_data <- read.csv("perception_study_data.csv", header=TRUE
)
```

```

perception._data$SYLL_MAXF0 <- scale(perception._data$SYLL_MAXF0)[,1]
perception._data$SYLL_MEAN_INT <- scale(perception._data$SYLL_MEAN_
  INT)[,1]
perception._data$SYLL_DUR <- scale(perception._data$SYLL_DUR)[,1]

max_form <- USER_RESP ~ PRIMARY_STRING + FUNCTION + EXPERIMENT +
  ONEBACK + SYLL_MAXF0 * SYLL_MEAN_INT * SYLL_DUR +
  (1 + PRIMARY_STRING + FUNCTION + ONEBACK +
  SYLL_MAXF0 * SYLL_MEAN_INT * SYLL_DUR | SUBJECT) +
  (1 + EXPERIMENT + ONEBACK | ITEM)

##### lme4
Full_Model_lme4 <- glmer(
  formula = max_form, data = perception._data, family = binomial)
# Model failed to converge with max|grad| = 0.0618226 (tol = 0.001,
  component 1)
# Time: 26 hours

Full_Model_bobyqa <- glmer(
  formula = max_form, data = perception._data, family = binomial,
  glmerControl(optimizer="bobyqa"))
# Model failed to converge with max|grad| = 0.00561514 (tol =
  0.001, component 1)

Full_Model_Nelder_Mead <- glmer(
  formula = max_form, data = perception._data, family = binomial,
  glmerControl(optimizer="Nelder_Mead"))
# Model failed to converge: degenerate Hessian with 2 negative
  eigenvalues

Intercepts_Only <- glmer(
  USER_RESP ~ PRIMARY_STRING + FUNCTION + EXPERIMENT + ONEBACK +
  SYLL_MAXF0 * SYLL_MEAN_INT * SYLL_DUR + (1|SUBJECT) + (1|ITEM),
  data = perception._data, family = binomial)
# Model failed to converge with max|grad| = 0.00746727 (tol =
  0.001, component 1)

```

```

One_Intercept<- glmer(
  USER_RESP ~ PRIMARY_STRING + FUNCTION + EXPERIMENT + ONEBACK +
  SYLL_MAXF0 * SYLL_MEAN_INT * SYLL_DUR + (1|SUBJECT) ,
  data = perception_data, family = binomial)
# Converges

##### stan
di <- getStanDataInfo(formula = max_form, data = perception_data,
  subj = "SUBJECT", item = "ITEM")
Full_Model_stan <- stan(file = "glmer_logistic.stan", data = di$
  data,
  chains = 3, iter = 2000, warmup = 1000, refresh = 100,
  pars = di$info$keep, open_progress = TRUE,
  sample_file = "perception_study.stancsv")
Full_Model_stan_summ <- getSummary(Full_Model_stan, di)
printSummary(Full_Model_stan_summ)
# divergent transitions
# Time: 5.5 hours

Full_Model_stan <- stan(file = "glmer_logistic.stan", data = di$
  data,
  chains = 3, iter = 2000, warmup = 1000, refresh = 100,
  pars = di$info$keep, open_progress = TRUE, control = list(adapt_
    delta = 0.99),
  sample_file = "perception_study.stancsv")
Full_Model_stan_summ <- getSummary(Full_Model_stan, di)
printSummary(Full_Model_stan_summ)
# Converges
# Time: 10 hours

```

9.3. Support Functions

```

##### printSummary
#####
##### Argument: list returned getSummary
##### Value: prints the major parts of the summary

printSummary <- function(summ){

```

```

w <- getOption("width")
options(width = 200)

cat("Mixed effects logistic regression fit with Stan\n\nFormula
:")
show(summ$formula)
cat("\n")

cat(paste("Observations:",summ$dims$N),
    paste("Unconstrained Parameters:",summ$dims$U),sep="\n")

y <- summ$response
if(any(y[-1]!=c("0","1"))){
  y <- paste(y[1]," (" ,paste(paste(y[-1],">",c("0","1")),
    collapse="; "),")",sep="")
} else y <- y[1]
cat(paste("Response:",y),"\n\n")

cat("Random Effects:\n\n")
for(r in 1:2){
  G <- ifelse(r==1,summ$dims$S,summ$dims$I)
  cat(paste(names(summ$random)[r]," (" ,G,")",sep=""),"\n")
  rsd <- format(round(summ$random[[r]]$sd,3),nsmall=3)
  rsc <- round(data.frame(summ$random[[r]]$cor),2)
  firstneg <- any(rsc[,1]<0)
  rsc <- format(rsc,nsmall=2)
  rsc[upper.tri(rsc,diag=T)] <- ""
  rsc <- cbind(names(rsd),rsd,rsc)
  rsc <- rsc[,-ncol(rsc)]
  colnames(rsc)[1:2] <- c("Name","SD")
  colnames(rsc)[3] <- ifelse(firstneg," Corr","Corr")
  if(ncol(rsc)>3) colnames(rsc)[4:ncol(rsc)] <- ""
  print(rsc,right=F,row.names=F)
  cat("\n")
}

cat("Fixed Effects:\n")
b <- summ$fixed$coef

```



```

b[,2:5] <- format(round(b[,2:5],3),nsmall=3)
p <- b[,6] <- round(b[,6],4)
b[,6] <- substr(format(b[,6],nsmall=4),2,6)
for(j in 2:5){
  if(any(round(summ$fixed$coef[,j],3)<0)){
    colnames(b)[j] <- paste(" ",colnames(b)[j],sep="")
  }
}
b[p==0,6] <- "<.0001"
b[p==1,6] <- ">.9999"
b[p!=0&p!=1,6] <- paste(" ",b[p!=0&p!=1,6],sep="")
colnames(b)[6] <- " P(B>0)"
print(b,row.names=F, right=F)

allpars <- do.call(rbind,args=summ$pars)
cat("", "Effect Sample Sizes:",sep="\n")
show(summary(allpars[, "n_eff"]))
cat("", "Rhat:",sep="\n")
show(summary(allpars[, "Rhat"]))

if(length(summ$messages)>0) cat("",summ$messages,sep="\n")

options(width = w)
}

#
#####

##### getStanDataInfo
#####
##### Arguments
### 'formula'
# A formula as would be passed to glmer()
# e.g. y ~ x1 + x2 + (1 + x1 + x2 | subject) + (1 + x1 + x2 | item)
# All numeric variables included in the formula should be scaled
# to mean 0 and standard deviation 1, and logicals
# (i.e. 0/1 predictors or T/F predictors) should be converted to

```

```

# factors prior to running the function. The function
  automatically
# sets sum contrasts for unordered factors and polynomial contrasts
# for ordered factors.
#
#### 'data'
# The data.frame which contains the data to apply the formula to.
#
#### 'subj'
# Character string indicating which of the two groups in the
  formula
# above corresponds to subjects, e.g. "subject", "speaker", etc.
# This is required so the names of random effects parameters in the
# output match correctly.
#
#### 'item'
# Character string indicating which of the two groups in the
  formula
# above corresponds to items, e.g. "item", "word", etc.
# This is required so the names of random effects parameters in the
# output match correctly.
#
##### Value: a list with two elements
#### 'data'
# A list which can be passed as the 'data' argument in stan() using
# the stan code in Kimball, Shantz, Eager, and Roy (2016).
#
#### 'info'
# A list which contains information about the names of the
  parameters
# that will be returned by stan(). It also contains a character
# vector 'keep' with the names of the transformed parameters which
# can be passed to the 'pars' argument in stan() to prevent the raw
# parameters from being returned as part of the model (saves space)

getStanDataInfo <- function(formula, data, subj, item){
  require(lme4)
  require(rstan)

```

```

require(stringr)
options(contrasts=c("contr.sum", "contr.poly"))

# get the response and fixed effects model matrix
fixd <- nobars(formula)
fr <- model.frame(fixd, data, drop.unused.levels=T)
y <- factor(model.response(fr))
ynames <- c(colnames(fr)[1], levels(y))
y <- as.numeric(y) - 1
x <- model.matrix(fixd, fr)
N <- nrow(x)
P <- length(Pnames <- colnames(x))

# get the random effects matrices
rand <- mkReTrms(findbars(formula), data)
S <- length(Snames <- levels(rand$flist[, subj]))
QS <- length(QSnames <- rand$cnms[[subj]])
I <- length(Inames <- levels(rand$flist[, item]))
QI <- length(QInames <- rand$cnms[[item]])

# in case interaction order is different between P/Q
Psplit <- lapply(str_split(Pnames, ":"), sort)
fixs <- which(!(QSnames %in% Pnames))
fixi <- which(!(QInames %in% Pnames))
for(i in fixs){
  j <- sort(str_split(QSnames[i], ":")[[1]])
  p <- which(vapply(Psplit, function(x){
    ifelse(length(x)==length(j),
      all(x==j), FALSE)}, logical(1)))
  if(length(p)!=1) stop("random effect not in fixed effects")
  QSnames[i] <- Pnames[p]
}
for(i in fixi){
  j <- sort(str_split(QInames[i], ":")[[1]])
  p <- which(vapply(Psplit, function(x){
    ifelse(length(x)==length(j),
      all(x==j), FALSE)}, logical(1)))
  if(length(p)!=1) stop("random effect not in fixed effects")

```

```

      QInames[i] <- Pnames[p]
    }

    OSnames <- expand.grid(QSnames, QSnames)[, 2:1]
    OInames <- expand.grid(QInames, QInames)[, 2:1]
    GSnames <- expand.grid(QSnames, Snames)[, 2:1]
    GInames <- expand.grid(QInames, Inames)[, 2:1]

    # combine fixed and random matrices and convert to CSR
    x <- extract_sparse_parts(cbind(x,
      t(rand$Ztlist[[which(names(rand$cnms)==subj)]]),
      t(rand$Ztlist[[which(names(rand$cnms)==item)]])))
    nz <- length(x$w)

    # return data and info
    return(list(
      data = list(N = N, S = S, I = I, P = P, QS = QS, QI = QI,
        y = y, nz = nz, x_w = x$w, x_v = x$v, x_u = x$u),
      info = list(formula = formula, subj = subj, item = item,
        y = ynames, P = Pnames, S = Snames, QS = QSnames,
        I = Inames, QI = QInames, OS = OSnames, OI = OInames,
        GS = GSnames, GI = GInames, keep = c("beta", "y_hat",
        "gamma_subj", "sigma_subj", "omega_subj", "gamma_item",
        "sigma_item", "omega_item"))))
  }

#
#####

##### getSummary
#####
##### Arguments
#### model
# The stanfit fit using data returned by getStanDataInfo and
# the stan code in Kimball, Shantz, Eager, and Roy (2016)
#
#### datainfo
# The list returned by getStanDataInfo

```

```

#
##### Value: a list with the following elements
### 'formula'
# The formula passed to getStanDataInfo
#
### 'response'
# A character vector with the name of the response variable
# and its levels
#
### 'fixed'
# A list with the fixed regression estimates which includes the
# posterior probability that each estimate is greater than zero,
# plus the fixed effect covariance and correlation matrices
#
### 'random'
# A list with a sub-list for subjects and a sub-list for items.
# Each list includes the estimates for individual subjs/items
# (ranef), the betas summed with these estimates (coef),
# the standard deviation in each effect (sd), and the effect
# covariance and correlation matrices (cov and cor)
#
### 'pars'
# A list of matrices, each being the summary for a given set
# parameters from stan's summary() function, but renamed to
# match the data provided to getStanDataInfo.
#
### 'parnames'
# A list that matches the names in the stanfit to the names
# in the summary, along with dimnames which can be assigned
# to extracted parameters

getSummary <- function(model, datainfo){
  d <- datainfo$data
  i <- datainfo$info

  # individual parameter summaries
  pn <- list(
    beta = list(stanname="beta",

```

```

      dimnames=list(NULL,i$P)),
gamma_subj = list(stanname="gamma_subj",
      dimnames=list(NULL,i$S,i$QS)),
sigma_subj = list(stanname="sigma_subj",
      dimnames=list(NULL,i$QS)),
omega_subj = list(stanname="omega_subj",
      dimnames=list(NULL,i$QS,i$QS)),
gamma_item = list(stanname="gamma_item",
      dimnames=list(NULL,i$I,i$QI)),
sigma_item = list(stanname="sigma_item",
      dimnames=list(NULL,i$QI)),
omega_item = list(stanname="omega_item",
      dimnames=list(NULL,i$QI,i$QI)),
predicted = list(stanname="y_hat",
      dimnames=list(NULL,paste(1:d$N)))
names(pn)[2:7] <- c(paste("gamma",i$subj,sep="_"),
      paste("sigma",i$subj,sep="_"),paste("omega",i$subj,sep="_"),
      ,
      paste("gamma",i$item,sep="_"),paste("sigma",i$item,sep="_"),
      ,
      paste("omega",i$item,sep="_"))

ps <- list()
for(p in names(pn)){
  sn <- pn[[p]]$stanname
  dn <- pn[[p]]$dimnames
  rn <- expand.grid(dn[length(dn):2])
  if(ncol(rn)>1) rn <- rn[,ncol(rn):1]
  rn <- apply(rn,1,function(x) paste(x,collapse=","))

  ps[[p]] <- summary(model,probs=c(.025,.975),pars=sn)$
    summary
  pn[[p]]$rownames <- data.frame(stan = rownames(ps[[p]]),
    named = paste(p,"[",rn,"]",sep=""), stringsAsFactors=F)
  rownames(ps[[p]]) <- pn[[p]]$rownames$named
}

# remove upper tri and diag from correlation matrices

```

```

for(o in c(4,7)){
  keep <- as.vector(lower.tri(diag(sqrt(nrow(ps[[o])))))
  ps[[o]] <- ps[[o]][keep,]
  if(!is.matrix(ps[[o]])){ # if there is only one slope
    ps[[o]] <- matrix(ps[[o]],1,7)
    rownames(ps[[o]]) <- pn[[o]]$rownames$named[3]
  }
}

# fixed
b <- extract(model, pars="beta")$beta
ppos <- apply(b,2,function(x) mean(x>0))
bcov <- var(b)
bcor <- cor(b)
dimnames(bcov) <- dimnames(bcor) <- list(i$P,i$P)
b <- cbind(i$P,data.frame(ps$beta[,c(1,3:5)]))
rownames(b) <- NULL
colnames(b) <- c("Name", "Estimate", "SD", "2.5%", "97.5%")
b[, "P(B>0)"] <- ppos
f <- list(coef = b, cov = bcov, cor = bcor)

# random
subj <- list()
subj$ranef <- matrix(ps[[2]][,1],d$S,d$QS,byrow=T,
  dimnames=list(i$S,i$QS))
subj$coef <- matrix(ps$beta[,1],d$S,d$P,byrow=T,
  dimnames=list(i$S,i$P))
subj$coef[,i$QS] <- subj$coef[,i$QS] + subj$ranef
subj$sd <- ps[[3]][,1]
names(subj$sd) <- i$QS
subj$cor <- matrix(1,d$QS,d$QS,dimnames=list(i$QS,i$QS))
subj$cor[lower.tri(subj$cor)] <- ps[[4]][,1]
subj$cor[upper.tri(subj$cor)] <- ps[[4]][,1]
subj$cov <- diag(subj$sd) %*% subj$cor %*% t(diag(subj$sd))
dimnames(subj$cov) <- dimnames(subj$cor)

item <- list()
item$ranef <- matrix(ps[[5]][,1],d$I,d$QI,byrow=T,

```

```

      dimnames=list(i$I,i$QI))
item$coef <- matrix(ps$beta[,1],d$I,d$P,byrow=T,
      dimnames=list(i$I,i$P))
item$coef[,i$QI] <- item$coef[,i$QI] + item$ranef
item$sd <- ps[[6]][,1]
names(item$sd) <- i$QI
item$cor <- matrix(1,d$QI,d$QI,dimnames=list(i$QI,i$QI))
item$cor[lower.tri(item$cor)] <- ps[[7]][,1]
item$cor[upper.tri(item$cor)] <- ps[[7]][,1]
item$cov <- diag(item$sd) %*% item$cor %*% t(diag(item$sd))
dimnames(item$cov) <- dimnames(item$cor)

r <- list(subj = subj, item = item)
names(r) <- c(i$subj,i$item)

# some diagnostics
ndiv <- sum(do.call(rbind, args = get_sampler_params(
      model, inc_warmup = FALSE))[, "divergent_"])
allpars <- do.call(rbind, args = ps)
rhat <- sum(allpars[, "Rhat"]>1.1)
neff <- sum(allpars[, "n_eff"]<100)
d$U <- 2*d$P+1 + (d$S+1)*d$QS+choose(d$QS,2) +
(d$I+1)*d$QI+choose(d$QI,2)

mess <- character()
if(ndiv>0) mess <- c(mess,paste("WARNING:",ndiv,
      "divergent transitions post-warmup"))
if(rhat>0) mess <- c(mess,paste("WARNING:",rhat,
      "parameters have Rhat values greater than 1.1"))
if(neff>0) mess <- c(mess,paste("WARNING:",neff,
      "parameters have effective sample sizes under 100"))
if(d$U>d$N) mess <- c(mess,paste("WARNING: model has more",
      "unconstrained parameters than observations"))

return(list(formula = i$formula, response = i$y, fixed = f,
      random = r, pars = ps, parnames = pn,
      dims = d[c("N","S","I","P","QS","QI","U")], messages = mess
    ))

```


}

#

#####